AudioCodes™ Mediant™ Media Gateways

SIP

Mediant 1000

User's Manual Version 5.0

Document #: LTRT-83301

December 2006

Table of Contents

List of Figures

List of Tables

Reader's Notes

Notices

Notice

This document describes the AudioCodes Mediant 1000 Voice-over-IP (VoIP) SIP media gateway.

Information contained in this document is believed to be accurate and reliable at the time of printing. However, due to ongoing product improvements and revisions, AudioCodes cannot guarantee accuracy of printed material after the Date Published nor can it accept responsibility for errors or omissions. Updates to this document and other documents can be viewed by registered Technical Support customers at http://www.audiocodes.com/ under Support / Product Documentation.

Date Published: Dec-07-2006 Date Printed: Dec-12-2006

Tip: When viewing this manual on CD, Web site or on any other electronic copy, all cross-references are hyperlinked. Click on the page or section numbers (shown in blue) to reach the individual cross-referenced item directly. To return back to the point from where you accessed the cross-reference, press the **ALT** and ◄ keys.

Trademarks

AC logo, Ardito, AudioCoded, AudioCodes, AudioCodes logo, IPmedia, Mediant, Mediant 1000, MP-MLQ, NetCoder, Stretto, TrunkPack, VoicePacketizer and VoIPerfect, are trademarks or registered trademarks of AudioCodes Limited. All other products or trademarks are property of their respective owners.

WEEE EU Directive

Pursuant to the WEEE EU Directive, electronic and electrical waste must not be disposed of with unsorted waste. Please contact your local recycling authority for disposal of this product.

Customer Support

Customer technical support and service are provided by AudioCodes' Distributors, Partners, and Resellers from whom the product was purchased. For Customer support for products purchased directly from AudioCodes, contact mailto:support@audiocodes.com.

Abbreviations and Terminology

Each abbreviation, unless widely used, is spelled out in full when first used. Only industry-standard terms are used throughout this manual. Hexadecimal notation is indicated by 0x preceding the number.

Related Documentation

Document #	Manual Name
LTRT-831xx (e.g., LTRT-83101)	Mediant 1000 SIP Release Notes
LTRT-835xx	Mediant 1000 MEGACO-H.323-SIP Fast Track Guide
LTRT-665xx	CPE SIP Configuration Guide for IP Voice Mail

Warning: Ensure that you connect FXS ports to analog telephone or to PBX-trunk lines only and FXO ports to CO/PBX lines only.

Warning: Disconnect the Mediant 1000 from the mains and from the Telephone Network Voltage (TNV) before servicing.

Warning: To protect against electrical shock and fire, use a 26 AWG min wire to connect FXO ports to the Public Switching Telephone Network (PSTN).

Warning: The FXO port is considered to be TNV-3. FXS ports are considered to be TNV-2.

Note: Where 'network' appears in this manual, it means Local Area Network (LAN), Wide Area Network (WAN), etc. accessed via the gateway's Ethernet interface.

Note: **FXO** (Foreign Exchange Office) is the interface replacing the analog telephone and connects to a Public Switched Telephone Network (PSTN) line from the Central Office (CO) or to a Private Branch Exchange (PBX). The FXO is designed to **receive** line voltage and ringing current, supplied from the CO or the PBX (just like an analog telephone). An FXO VoIP gateway interfaces between the CO/PBX line and the Internet.

FXS (Foreign Exchange Station) is the interface replacing the Exchange (i.e., the CO or the PBX) and connects to analog telephones, dial-up modems, and fax machines. The FXS is designed to **supply** line voltage and ringing current to these telephone devices. An FXS VoIP gateway interfaces between the analog telephone devices and the Internet.

Note: Throughout this manual, the term 'Trunk' is used synonymously with 'Hunt'. Trunk typically refers to digital modules while Hunt typically refers to analog modules

1 Overview

The AudioCodes Mediant 1000 is a best-of-breed Voice-over-IP (VoIP) SIP media gateway, using field-proven, market-leading technology, implementing analog and digital cutting-edge technology. The Mediant 1000 is designed to seamlessly interface between TDM and IP networks, providing superior voice quality and optimized packet voice streaming (voice, fax, and data traffic) over IP networks.

The Mediant 1000 is best suited for small-to-medium size (SME) enterprises, branch offices, or for residential media gateway solutions. The Mediant 1000 is a highly scalable and modular system that matches the density requirements for smaller environments, while meeting service providers' demands for growth.

The Mediant 1000 is ideal for connecting an enterprise's legacy telephones, fax machines and PBX systems to IP-based telephony networks, as well as for seamless connection of IP-based PBX architecture to the PSTN. In addition to operating as a pure media gateway, the Mediant 1000 open platform extends its flexibility with additional deployment options to host partner applications, known as the Open Solutions Network (OSN) Server for supporting third-party VoIP applications such as IP-PBX, Pre-Paid, and IP-PBX redundancy.

The Mediant 1000 also provides conferencing services over VoIP networks. This is supported by an optional Conference module that can be housed in the Mediant 1000 chassis.

The Mediant 1000 is fully interoperable with multiple vendor gateways, softswitches, SIP servers, gatekeepers, proxy servers, IP phones, session border controllers, and firewalls. The Mediant 1000 is designed to meet NEBS Level 3 (Bellcore) and regulatory approval (including Safety, EMC, and Telecom for USA, EU and other countries).

Intelligently packaged in a stackable 1U chassis, the Mediant 1000 gateways are very compact devices that can be mounted as desk-top units, on the wall, or in standard 19-inch racks. The Mediant 1000 is provided with two integral mounting brackets for facilitating the rack installation.

Mediant 1000 units are equipped with two 10/100 Base-TX Ethernet ports for connection to the IP network. The second Ethernet port is used for 1+1 Ethernet redundancy.

The Mediant 1000 supports mixed digital and analog interface configurations, providing up to four E1/T1/J1 spans and up to 24 analog interfaces in various FXO/FXS configurations.

The Mediant 1000 digital interface supports multiples of 1, 2, or 4 E1/T1/J1 spans used for connecting the PSTN or PBX to the IP network. The digital modules provide RJ-48 ports. The digital module can be configured as regular E1/T1/J1 interfaces, and with up to 1 or 2 paired spans acting as lifeline telephone interfaces for switching to the PSTN in case of power failure or network problems.

The Mediant 1000 analog interface supports up to 24 analog ports (four ports per module) in various Foreign Exchange Office (FXO) or Foreign Exchange Station (FXS) configurations, supporting up to 24 simultaneous VoIP calls. Each analog module comprises four analog RJ-11 ports. The FXO module can be used to connect analog lines of an enterprise's PBX or of the PSTN to the IP network. The FXS module can be used to connect legacy telephones, fax machines, and modems to the IP network. Optionally, the FXS module can be connected to the external trunk lines of a PBX. When deployed with a combination of FXO and FXS modules, the Mediant 1000 can be used as a PBX for Small Office Home Office (SOHO) users, and businesses not equipped with a PBX.

The Mediant 1000 has enhanced hardware and software capabilities to ease its installation and to help maintain voice quality. If the measured voice quality falls beneath a pre-configured value, or the path to the destination is disconnected, the Mediant 1000 can assure voice connectivity by falling back to the PSTN. In the event of network problems or power failures, calls can be routed back to the PSTN without requiring routing modifications in the PBX. Further reliability is provided by dual Ethernet ports and optional dual AC power supply.

The Mediant 1000 supports various ISDN PRI protocols such as EuroISDN, North American NI2, Lucent™ 4/5ESS, Nortel™ DMS100 and others. In addition, it supports different variants of CAS protocols for E1 and T1 spans, including MFC R2, E&M immediate start, E&M delay dial / start, loop start and ground start.

The Mediant 1000 provides a user-friendly embedded HTTP-based Web server for remote configuration and management using a standard Web browser (such as Microsoft™ Internet Explorer™ or Netscape™ Navigator™), from anywhere in the world with IP connectivity to the device.

1.1 SIP Overview

SIP (Session Initialization Protocol) is an application-layer control (signaling) protocol used on the Mediant 1000 for creating, modifying, and terminating sessions with one or more participants. These sessions can include Internet telephone calls, media announcements and conferences.

SIP invitations are used to create sessions and carry session descriptions that enable participants to agree on a set of compatible media types. SIP uses elements called Proxy servers to help route requests to the user's current location, authenticate and authorize users for services, implement provider call-routing policies and provide features to users.

SIP also provides a registration function that enables users to upload their current locations for use by Proxy servers. SIP, on the Mediant 1000, complies with the IETF (Internet Engineering Task Force) RFC 3261 (refer to http://www.ietf.org/rfc/rfc3261.txt?number=3261).

1.2 Mediant 1000 Features

This section provides a high-level overview of some of the many Mediant 1000 supported features.

1.2.1 General Features

- Superior, high quality Voice, Data and fax over IP networks.
- Toll quality voice compression.
- Enhanced capabilities including MWI, long haul, metering, CID and out door protection.
- Proven integration with leading PBXs, IP-PBXs, Softswitches and SIP servers.
- Supports multiples of 1, 2, or 4 E1/T1/J1 spans
- Spans a range of 4 to 24 FXS/FXO analog ports (in multiples of 4).
- Selectable G.711 or multiple Low Bit Rate (LBR) coders per channel.
- T.38 fax with superior performance (handling a round-trip delay of up to nine seconds).
- Echo Canceler, Jitter Buffer, Voice Activity Detection (VAD) and Comfort Noise Generation (CNG) support.
- Comprehensive support for supplementary services.
- Web Management for easy configuration and installation.
- Simple Network Management Protocol (SNMP) and Syslog support.
- ThroughPacket™ proprietary feature that aggregates payloads from several channels into a single IP packet to reduce bandwidth overhead.
- Supports load balancing with proxy.
- T.38 fax fallback to PCM (or NSE).
- Can be integrated into a Multiple IPs and a VLAN-aware environment.
- Capable of automatically updating its firmware version and configuration.

- Secured Web access (HTTPS) and Telnet access using SSL / TLS.
- IPSec and IKE protocols are used in conjunction to provide security for control (e.g., SIP) and management (e.g., SNMP and Web) protocols.

1.2.2 PSTN-to-SIP Interworking

The Mediant 1000 Digital SIP gateway supports various ISDN PRI protocols such as EuroISDN, North American NI2, Lucent 4/5ESS, Nortel DMS100, Meridian1 DMS100, Japan J1, as well as QSIG. PRI support includes User Termination or Network Termination side. ISDN-PRI protocols can be defined on an E1/T1 basis (i.e., different variants of PRI are allowed on different E1/T1 spans).

In addition, the gateway supports numerous variants of CAS protocols for E1 and T1 spans, including MFCR2, E&M wink start, E&M immediate start, E&M delay dial/start, loop-start, and ground start. CAS protocols can be defined on an E1/T1 basis (i.e., different variants of CAS are allowed on different E1/T1 spans).

The gateway simultaneously supports different variants of CAS and PRI protocols on different E1/T1 spans (no more than four simultaneous PRI variants).

PSTN to SIP and SIP to PSTN Called and Calling numbers can be optionally modified according to rules that are defined in gateway's ini file.

1.2.3 Supported Interworking Features

- Definition and use of Trunk Groups for routing IP-to-PSTN calls.
- B-channel negotiation for PRI spans.
- ISDN Non Facility Associated Signaling (NFAS).
- PRI to SIP interworking according to <draft-ietf-sipping-qsig2sip-04.txt>.
- PRI to SIP Interworking of Q.931 Display (Calling name) information element.
- PRI (NI-2, 5ESS) to SIP interworking of Calling Name using Facility IE in Setup and Facility messages.
- Configuration of Numbering Plan and Type for IP-to-ISDN calls.
- Interworking and flexible mapping of PSTN to SIP release causes.
- Interworking of ISDN redirect number to SIP diversion header (according to IETF <draft-levy-sip-diversion-05.txt>).
- Optional change of redirect number to called number for ISDN-to-IP calls.
- Interworking of ISDN calling line Presentation & Screening indicators using RPID header <draft-ietf-sip-privacy-04.txt>.
- Interworking of Q.931 Called and Calling Number Type and Number Plan values using the RPID header.
- Interworking of Calling and Called Subaddress values for SIP-to-ISDN calls.
- Supports ISDN en-block or overlap dialing for incoming Tel-to-IP calls.
- Supports a digit map pattern to reduce the dialing period when Overlap dialing is used.
- Supports routing of IP-to-Tel calls to predefined trunk groups.
- Supports a configurable channel select mode per trunk group.
- Supports various number manipulation rules for IP-to-Tel and Tel-to-IP, called and calling numbers.
- Option to configure ISDN Transfer Capability per trunk.
- Supports transfer of User-to-User Information Element (UUIE) between SIP and PRI.
- Supports ISDN Transfer using TBCT / RLT / ECT.
- Supports QSIG messages tunneling over SIP according to <draft-elwell-sipping-qsig-tunnel-03>.

- Supports ISDN PRI messages tunneling over SIP messages according to RFC 3372 – SIP-T.
- Interworking of Calling Name, Calling Name Presentation, and Redirect Number for QSIG-to-SIP and SIP-to-QSIG calls.
- Supports QSIG Call Re-Route.
- Supports QSIG MWI Notifications.

1.2.4 Supported Media Server Features

- VoIP Conference Bridge supporting up to 60 channels.
- Up to 60 participants in a full duplex conference.
- Up to 20 three-way conferences.

1.2.5 SIP Features

The Mediant 1000 SIP gateway complies with the IETF RFC 3261 standard.

- Reliable User Datagram Protocol (UDP) transport, with retransmissions.
- Transmission Control Protocol (TCP) Transport layer.
- SIPS using TLS.
- T.38 real time Fax (using SIP).

  Note: If the remote side includes the fax maximum rate parameter in the SDP body of the INVITE message, the gateway returns the same rate in the response SDP.
- Works with Proxy or without Proxy, using an internal routing table.
- Fallback to internal routing table if Proxy is not responding.
- Supports up to four Proxy servers. If the primary Proxy fails, the gateway automatically switches to a redundant Proxy.
- Supports domain name resolving using DNS NAPTR and SRV records for Proxy, Registrar and domain names that appear in the Contact and Record-Route headers.
- Supports Load Balancing over Proxy servers using Round Robin or Random Weights.
- Proxy or Registrar Registration such as:

  ```
  REGISTER sip:servername SIP/2.0
  VIA: SIP/2.0/UDP 212.179.22.229;branch=z9hG4bRaC7AU234
  From: <sip:GWRegistrationName@sipgatewayname>;tag=1c29347
  To: <sip:GWRegistrationName@sipgatewayname>
  Call-ID: 10453@212.179.22.229
  Seq: 1 REGISTER
  Expires: 3600
  Contact: sip:GWRegistrationName@212.179.22.229
  Content-Length: 0
  ```

  The "servername" string is defined according to the following rules:

  - The "servername" is equal to "RegistrarName" if configured. The "RegistrarName" can be any string.
  - Otherwise, the "servername" is equal to "RegistrarIP" (either FQDN or numerical IP address), if configured.
  - Otherwise the "servername" is equal to "ProxyName" if configured. The "ProxyName" can be any string.
  - Otherwise the "servername" is equal to "ProxyIP" (either FQDN or numerical IP address).

  The parameter GWRegistrationName can be any string. If the parameter is not defined, the parameter UserName is used instead.

The "sipgatewayname" parameter (defined in the ini file or set from the Web browser), can be any string. Some Proxy servers require that the "sipgatewayname" (in REGISTER messages) is set equal to the Registrar/Proxy IP address or to the Registrar/Proxy domain name.

The REGISTER message is sent to the Registrar's IP address (if configured) or to the Proxy's IP address. The message is sent per gateway or per gateway endpoint according to the "AuthenticationMode" parameter. Usually the FXS gateways are registered per gateway port, while FXO gateways send a single registration message, where Username is used instead of phone number in From/To headers. The registration request is resent according to the parameter 'RegistrationTimeDivider'. For example, if 'RegistrationTimeDivider = 70' (%) and Registration Expires time = 3600, the gateway resends its registration request after 3600 x 70% = 2520 sec. The default value of 'RegistrationTimeDivider' is 50%.

If registration per B-channel is selected, on gateway startup, the gateway sends REGISTER requests according to the maximum number of allowed SIP dialogs (configured by the parameter NumberOfActiveDialogs). After each received response, the subsequent REGISTER request is sent.

- Proxy and Registrar Authentication (handling 401 and 407 responses) using Basic or Digest methods. Accepted challenges are kept for future requests to reduce the network traffic.
- Single gateway Registration or multiple Registration of all gateway endpoints.
- Supported methods: INVITE, CANCEL, BYE, ACK, REGISTER, OPTIONS, INFO, REFER, UPDATE, NOTIFY, PRACK, SUBSCRIBE and PUBLISH.
- Modifying connection parameters for an already established call (re-INVITE).
- Working with Redirect server and handling 3xx responses.
- Early media (supporting 183 Session Progress).
- PRACK reliable provisional responses (RFC 3262).
- Call Hold and Transfer Supplementary services using REFER, Refer-To, Referred-By, Replaces and NOTIFY.
- Supports RFC 3711, Secured RTP and Key Exchange according to <draft-ietf-mmusic-sdescriptions-12>.
- Supports RFC 3489, Simple Traversal of UDP Through NATs (STUN).
- Supports RFC 3605, RTCP Attribute in SDP.
- Supports RFC 3326, Reason header.
- Supports RFC 3911, The SIP Join Header.
- Supports RFC 3903, SIP Extension for Event State Publication.
- Supports RFC 3953, The Early Disposition Type for SIP.
- Supports for RFC 3966, The tel URI for Telephone Numbers.
- Supports RFC 4244, An Extension to SIP for Request History Information.
- Supports ITU V.152 - Procedures for supporting Voice-Band Data over IP Networks.
- Supports RFC 3327, Adding 'Path' to Supported header.
- Supports RFC 3581, Symmetric Response Routing.
- Supports RFC 4028, Session Timers in SIP.
- Supports network asserted identity and privacy (RFC 3325 and RFC 3323).
- Supports Tel URI (Uniform Resource Identifier) according to RFC 2806 bis.
- Remote party ID <draft-ietf-sip-privacy-04.txt>.
- Supports obtaining Proxy Domain Name(s) from DHCP (Dynamic Host Control Protocol) according to RFC 3361.
- Supports handling forking proxy multiple responses.
- RFC 2833 Relay for DTMF Digits, including payload type negotiation.

- DTMF out-of-band transfer using:
    - INFO method <draft-choudhuri-sip-info-digit-00.txt>.
    - INFO method, compatible with Cisco gateways.
    - NOTIFY method <draft-mahy-sipping-signaled-digits-01.txt>.
- SIP URL: sip:"phone number"@IP address (such as 122@10.1.2.4, where "122" is the phone number of the source or destination phone number) or sip:"phone_number"@"domain name", such as 122@myproxy.com. Note that the SIP URI host name can be configured differently per called number.
- Supports RFC 4040, RTP payload format for a 64 kbit/s transparent data.
- Can negotiate coder from a list of given coders.
- Supports negotiation of dynamic payload types.
- Supports multiple ptime values per coder.
- Supports RFC 3824, Using E.164 numbers with SIP (ENUM).
- Supports RFC 3389, RTP Payload for Comfort Noise.
- Supports reception and DNS resolution of FQDNs received in SDP.
- Supports <draft-ietf-sip-gruu-09>, Obtaining and Using Globally Routable User Agent (UA) URIs (GRUU) in SIP
- Implementation of MWI IETF <draft-ietf-sipping-mwi-04.txt>, including SUBSCRIBE (to the MWI server). The FXS gateways can accept an MWI NOTIFY message that indicates waiting messages or indicates that the MWI is cleared.
- Representing hunt/trunk groups in tel/sip Uniform Resource Identifiers (URIs) according to <draft-ietf-iptel-trunk-group-04>.
- Supports RTCP-XR reports publishing according to RFC 3611 and draft-johnston-sipping-rtcp-summary-07.
- Responds to OPTIONS messages both outside a SIP dialog and in mid-call. Generates SIP OPTIONS messages as Proxy keep-alive mechanism.
- Publishes the total number of free Tel channels in a 200 OK response to an OPTIONS requests.
- Supports 3-Way Conference using an external media server.

For more updated information on the gateway's supported features, refer to the latest Mediant 1000 SIP Release Notes.

2 Physical Description

Designed to meet Network Equipment Building System (NEBS) Level 3, the Mediant 1000 is a 19-inch industrial platform chassis, 1U high and 13.8 inch deep.

The Mediant 1000 supports a scalable, modular architecture that includes up to six analog modules, up to four digital modules, a single CPU module, a power supply module and an optional fan try module (the extractable modules are shown in Section 2.3 on page 35).

This section provides detailed information on the:

- Mediant 1000 front panel (refer to Section 2.1 on page 29).
- Mediant 1000 rear panel (refer to Section 2.2on page 34).
- Mediant 1000 modules (refer to 2.3 on page 35).

2.1 Mediant 1000 Front Panel

Figure 2-1 shows a front view of the Mediant 1000.

Figure 2-1: Mediant 1000 Front View and CPU Enlargement

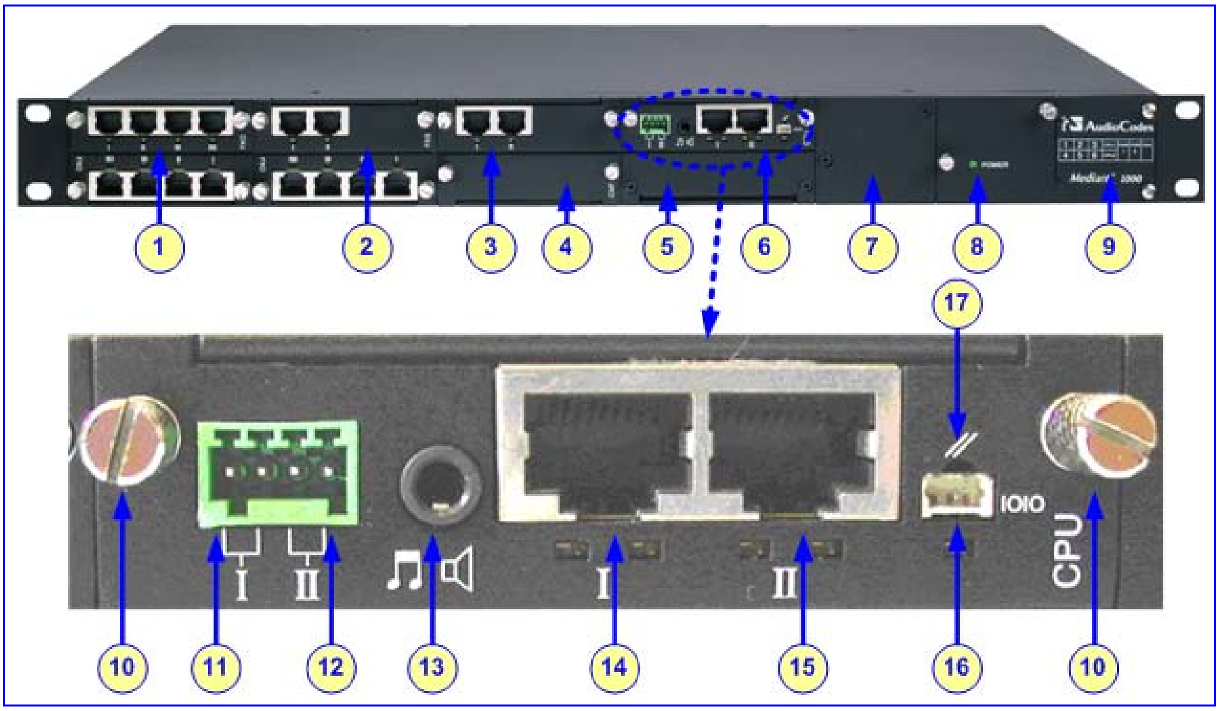


Table 2-1: Mediant 1000 Front View Component Descriptions (continues on pages 29 to 30)

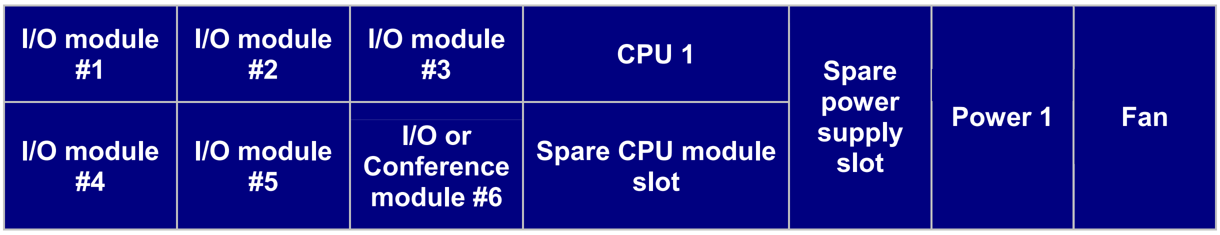
Item #	Label	Component Description
Front View of Chassis		
1	**FXO**	4-port FXO (or FXO G) module.
2	**FXS**	2-port FXS module.
3	**TRUNKS**	2 RJ-48c ports digital module (E1/T1/J1)
4	**CNF**	Conference module.

Table 2-1: Mediant 1000 Front View Component Descriptions (continues on pages 29 to 30)

Item #	Label	Component Description
5	**CPU**	Spare CPU module slot and also used for OSN server where slot hosts connection module (for OSN Server installation, refer to Appendix K on page 465).
6	**CPU**	Main CPU module.
7	**Power 1**	Spare power supply slot.
8	**Power 2**	Main power supply.
9	**Schematic**	Extractable fan tray.
		Enlarged View of CPU
10	**-**	Locking screws (2).
11	**I**	Dry contact port (normally open)
12	**II**	Dry contact port (normally closed)
13	♫	Audio IN/OUT (for paging and MOH (Music on Hold) functionalities).
14	**I**	10/100 Base-TX Ethernet Port 1.
15	**II**	10/100 Base-TX Ethernet Port 2.
16	**I0I0**	RS-232 port.
17	**//**	Reset button.

Figure 2-2 illustrates the front layout of the Mediant 1000. There is also a schematic of the front layout on the front panel of the fan tray. To view your specific device's configuration using the Embedded Web Server, refer to Section 5.10 on page 202.

Figure 2-2: Mediant 1000 Front Layout



I/O module #1	I/O module #2	I/O module #3	CPU 1	Spare power supply slot	Power 1	Fan
I/O module #4	I/O module #5	I/O or Conference module #6	Spare CPU module slot			

Note: The I/O modules must be housed in consecutive slots. In other words, if the Mediant 1000 houses three I/O modules, they have to occupy slots 1, 2, and 3.

2.1.1 I/O Modules

The Mediant 1000 can house both analog and or digital modules:

- **Analog modules:** the gateway supports up to six replaceable analog FXO and/or FXS modules. Each module contains four analog RJ-11 ports. Therefore, the gateway can support up to 24 analog ports (6 modules x 4 ports).
- **Digital modules:** the gateway supports up to four digital trunks (fully flexible, from a single up to four trunks per module). The digital modules are available in 1, 2, or 4 spans. If the power fails, a relay connects trunks 1 to 2, and 3 to 4 (in the same module) acting as a fallback for PSTN trunks.

Note: The standard FXO modules support outdoor and indoor (lightning protection) loop start signaling. For ground start signaling, the **FXO G** modules are required. These modules support loop and ground start, and only support indoor protection. (The FXS modules support both loop and ground start signaling.)

To enable ground start, use the *ini* file parameter GroundKeyDetection (refer to Section 6.5.2 on page 220).

2.1.2 CPU Module

The CPU (Central Processing Unit) module is located to the right of the six I/O analog/digital modules.

Figure 2-1 shows the CPU module's front panel ports and connectors described in sequence from left to right in the following sections.

2.1.2.1 Dry Contact Connector (Labeled I and II)

The Mediant 1000 provides dry contacts that can be connected to an external audible or visual alarm system (bell, siren, hooter, or light).

2.1.2.2 Audio IN/OUT[1]

The Audio IN/OUT port is indicated by the musical note and loudspeaker symbols (refer to Figure 2-1 on page 29). It is used for Music on Hold (IN) and paging (OUT).

2.1.2.3 10/100 Base-TX Ethernet Ports (Labeled I and II)

Two 10/100 Base-TX Ethernet ports provide a dual Ethernet redundancy scheme, protecting against failure (for example, a disconnection) of any cable or associated LAN switch port.

[1] Not currently supported. Contact AudioCodes regarding availability.

2.1.2.4 RS-232 Port (Labeled I0I0)

The RS-232 port is used to access the CLI (Section 4.2.4 on page 50) and to receive error / notification messages (a 9-pin DB adapting cable is supplied).

Note: The RS-232 port is not intended for a permanent connection.

2.1.2.5 Reset Button (Labeled //)

The Mediant 1000 Reset button is located directly above the RS-232 port.

To reset the system, take a pointed object and press in the Reset button.

You can also use the Reset button to restore the Mediant 1000 networking parameters to their factory default values (refer to Section 12.1 on page 319).

2.1.3 Conference Module

The Conference module is used for media server support (i.e., conferencing). This module is installed in slot 6 of the chassis front panel. For a description of Mediant 1000 conferencing capabilities, refer to Section 8 on page 263.

2.1.4 Power Supply Module (Labeled 1 and 2)

The Mediant 1000 features two extractable power supply units: Power 1 and Power 2 (refer to Figure 2-2 on page 30), with an AC power connector at the rear of each (refer to Figure 2-4 on page 34).

If both Power 1 and Power 2 units are used, they share the load between them. This (optional) load-sharing feature enables failure protection / redundancy. When using this feature, you are advised to connect each power supply unit to a different AC supply circuit.

The front panel of the power supply unit provides a power supply LED that is lit green when the Mediant 1000 is powered up. If this LED does not light up, a power supply problem may be present.

2.1.5 Fan Tray Module

The Mediant 1000 components are cooled by a fan tray unit located to the extreme right when facing the front panel (refer to Figure 2-2 on page 30).

The fan tray unit draws in air through a perforated grill at the right side of the chassis. The incoming air passes through a removable filter, whose honeycombed design prevents RF interference. The clean air passes through the entire set of modules cooling each one and exits the Mediant 1000 via perforated vents on the left side of the chassis.

Blank panels are used to cover all unoccupied slots on the front and rear sides of the chassis. The blank panels are especially constructed to assist optimal air flow within the chassis.

For replacing the Fan tray unit, refer to Section 3.5.3 on page 45.

Note: It is imperative to cover all unoccupied slots in the front and rear panels of the chassis with blank panels to maintain internal airflow pressure.

2.1.6 Front Panel LEDs

Figure 2-3 below, shows the location of the front panel LEDs on the Mediant 1000. They are described below in Table 2-2, Table 2-3, Table 2-4, and Table 2-5.

Figure 2-3: Location of Front Panel LEDs

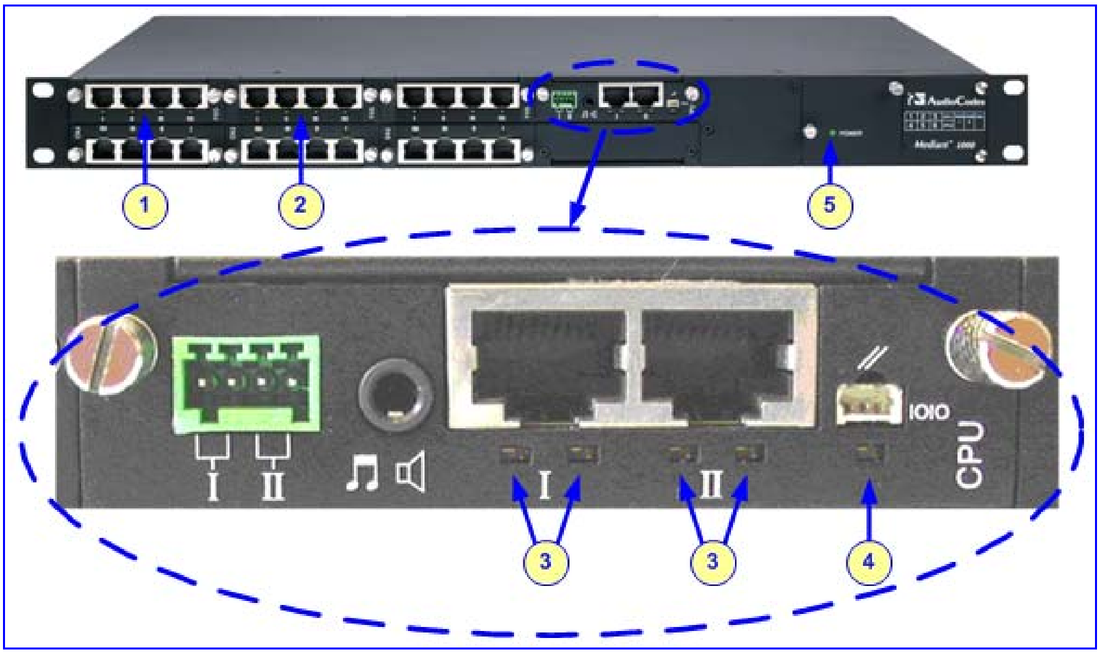


Table 2-2: CPU Module LEDs Description

LED	Location	Color	State	LED Indication
Ethernet Ports I & II	3 (Left LED)	**Orange**	Blinking	Activity.
	3 (Right LED)	**Green**	On	Link OK.
		Yellow	Blinking	Data is being received.
		---	Off	No link.
General Purpose	4	**Green**	N/A. (Future support.)	

Table 2-3: Analog I/O Modules LEDs Description

LED	Location	Color	State	LED Indication
RJ-11 (Only for analog interfaces)	1	**Green**	On	The LED is lit green when the phone is offhooked (FXS) or when the FXO offhooks the line towards the PBX.
			Blinking	The LED blinks green when the FXS rings the extension line or the FXO detects a ring signal from the PBX.
		Red	On	Error. The line is malfunctioning.

Table 2-4: Digital I/O Modules LED Description

LED	Location	Color	State	LED Indication
RJ-48c (Only for digital interfaces)	2	Green	On	Trunk is synchronized (normal operation)
		Red	On	Loss due to any of the following 4 signals: ▪ **LOS** - Loss of Signal ▪ **LOF** - Loss of Frame ▪ **AIS** - Alarm Indication Signal (the Blue Alarm) ▪ **RAI** - Remote Alarm Indication (the Yellow Alarm)
		-	Off	Either there is a failure / disruption in the AC power supply or the power is currently not being supplied to the Mediant 1000 through the AC power supply entry.

Table 2-5: Power Supply Module LED Description

LED	Location	Color	State	LED Indication
Power Supply	5	Green	On	The LED of each AC power supply is lit green when the power supply is operating correctly.
		-	Off	Either there is a failure / disruption in the AC supply or the power is currently not being supplied to the Mediant 1000 through the AC power supply entry.

2.2 Mediant 1000 Rear Panel

The Mediant 1000 rear panel provides the power connectors, as described in Table 2-6 (refer to Figure 2-4).

Figure 2-4: Mediant 1000 Rear Connectors

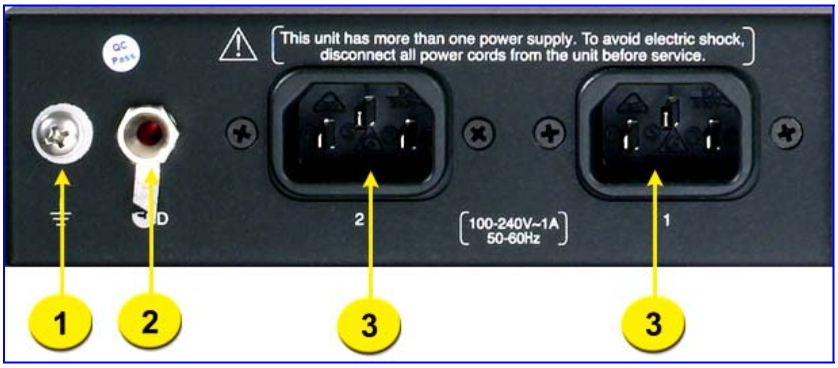

QC Pass
This unit has more than one power supply. To avoid electric shock, disconnect all power cords from the unit before service.
D
2
100-240V~1A 50-60Hz
1
1
2
3
3

Table 2-6: Mediant 1000 Rear Connectors Component Descriptions

Item #	Label	Component Description
1	⏚	Protective earthing screw.
2	ESD	Electrostatic Discharge (ESD) port.
3	100-240V~1A	Dual AC Power Supply Entry.

Note: The rear panel also provides module slots for housing OSN Server modules (viz., OSN Server and Hard Drive modules). For information on the OSN Server installation, refer to Appendix K on page 465.)

2.3 Mediant 1000 Modules

The extractable Mediant 1000 modules are shown in Table 2-7 below.

Table 2-7: Extractable Mediant 1000 Modules

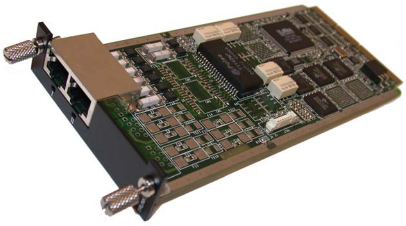
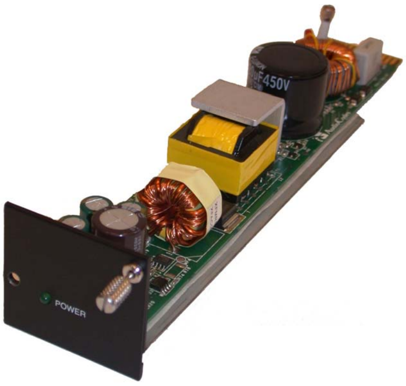
Module	Description
	CPU module
	Digital I/O module (e.g., 2 spans)
	4-port FXS Analog I/O module
	4-port FXO Analog I/O module
	4-port FXO G Analog I/O module (for ground start support)
	Conference module

Table 2-7: Extractable Mediant 1000 Modules

Module	Description
	Fan tray module with six fans and an air filter
	Power supply module

3 Installing the Mediant 1000

This section provides information on the hardware installation procedure for the Mediant 1000.

Caution Electrical Shock

The equipment must only be installed or serviced by qualified service personnel.

➢ To install your Mediant 1000, take these 4 steps:

1. Unpack the Mediant 1000 (refer to Section 3.1 below).
2. Check the package contents (refer to Section 3.2 below).
3. Mount the Mediant 1000 (refer to Section 3.3 on page 38).
4. Cable the Mediant 1000 (refer to Section 3.4 on page 39).

After connecting the Mediant 1000 to the power source, the power LED on the front panel of the power supply unit is lit green (after a self-testing period of about 2 minutes). Any power supply malfunction results in the LED switching off (refer to Table 2-2 on page 33 for details on the Mediant 1000 LEDs).

When you have completed the above relevant sections you are then ready to start configuring the gateway (Chapter 5 on page 44).

3.1 Unpacking

➢ To unpack the Mediant 1000, take these 6 steps:

1. Open the carton and remove packing materials.
2. Remove the Mediant 1000 from the carton.
3. Check that there is no equipment damage.
4. Check, retain and process any documents.
5. Notify AudioCodes or your local supplier of any damage or discrepancies.
6. Retain any diskettes or CDs.

3.2 Package Contents

Ensure that in addition to the Mediant 1000, the package contains:

- One or two AC power cables.
- Four anti-slide bumpers for desktop installation option (supplied in a small plastic bag).
- CD (software and documentation).
- RS-232 DB9 adapting cable, two meters in length (direct connection to PC).
- The Mediant 1000 Fast Track Installation Guide.

3.3 Mounting the Mediant 1000

The Mediant 1000 can be mounted on a desktop, or installed in a standard 19-inch rack. For detailed information on cabling the Mediant 1000, refer to Section 3.4 on page 39.

3.3.1 Mounting the Mediant 1000 on a Desktop

The integral front brackets are not required for a desktop installation. Optionally, attach the four (supplied) anti-slide bumpers to the base of the Mediant 1000 and place it on the desktop in the position you require.

3.3.2 Installing the Mediant 1000 in a 19-inch Rack

Users can install the Mediant 1000 in a standard 19-inch rack either by placing it on a shelf preinstalled in the rack (preferred method), or by attaching it directly to the rack's frame. If the rack is without shelves, it is advised to acquire shelves.

Installation of a Mediant 1000 in a rack without shelves requires users to directly attach the Mediant 1000 integral front brackets *and* the user-adapted rear brackets (not provided) to the rack's frame.

Note: Users assembling the rear brackets by themselves should note the following:

- The distance between the screws on each bracket is 28 mm.
- To attach the brackets, use 4-40 screws with a maximal box penetration length of 3.5 mm.

Rack Mount Safety Instructions (UL)

When installing the chassis in a rack, be sure to implement the following safety instructions recommended by Underwriters Laboratories:

- **Elevated Operating Ambient** - If installed in a closed or multi-unit rack assembly, the operating ambient temperature of the rack environment may be greater than room ambient. Therefore, consideration should be given to installing the equipment in an environment compatible with the maximum ambient temperature (Tma) specified by the manufacturer.
- **Reduced Air Flow** - Installation of the equipment in a rack should be such that the amount of air flow required for safe operation on the equipment is not compromised.
- **Mechanical Loading** - Mounting of the equipment in the rack should be such that a hazardous condition is not achieved due to uneven mechanical loading.
- **Circuit Overloading** - Consideration should be given to the connection of the equipment to the supply circuit and the effect that overloading of the circuits might have on overcurrent protection and supply wiring. Appropriate consideration of equipment nameplate ratings should be used when addressing this concern.
- **Reliable Earthing** - Reliable earthing of rack-mounted equipment should be maintained. Particular attention should be given to supply connections other than direct connections to the branch circuit (e.g., use of power strips.)

➢ To place the Mediant 1000 on a 19-inch rack's shelf, take these 2 steps:

1. Place the Mediant 1000 on a preinstalled shelf.
2. It is recommended to attach the Mediant 1000 integral front brackets to the frame of the rack to prevent it from sliding off the shelf during cabling. Use standard 19-inch rack bolts (not provided) to fasten the front of the Mediant 1000 to the frame of the rack.

➢ To install the Mediant 1000 in a rack without shelves, take these 2 steps:

1. Position the Mediant 1000 in your 19-inch rack and align the *front and rear* bracket holes to holes (of your choosing) in the vertical tracks of the 19-inch rack.
2. Use standard 19-inch rack bolts (not provided) to fasten the brackets to the frame of the rack.

3.4 Cabling the Mediant 1000

Protective Earthing

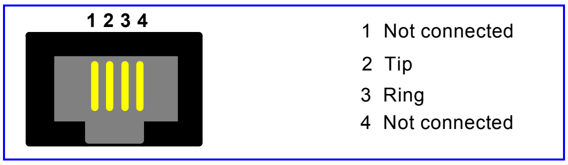
The equipment is classified as Class I EN60950 and UL60950 and must be earthed at all times.

To protect against electrical shock and fire, use a 26 AWG min wire to connect FXO, T1, or, E1 ports to the PSTN.

➢ To cable the Mediant 1000, take these 6 steps:

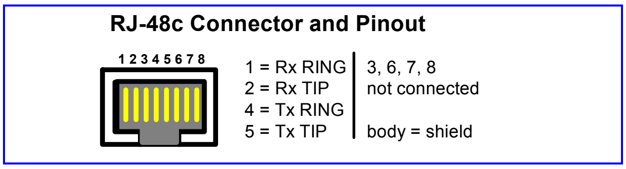
1. The Mediant 1000 must be permanently earthed using an equipment-earthing conductor. Connect an electrically earthed strap of 16 AWG wire (minimum) to the chassis earthing screw, using the supplied washer. The connection to the protective earthing should be in accordance with the regulations enforced in the country of installation.
2. For cabling the analog and digital I/O module interfaces:
   - **Analog interfaces:** Connect the RJ-11 connectors of the analog FXS module/s to a fax machine, modem or phones. Connect the RJ-11 connectors of the analog FXO module/s to telephone exchange analog lines or PBX extensions. Ensure that FXS and FXO are connected to the correct external devices, otherwise damage can occur. Refer to the figure below for the RJ-11 telephone connector pinout.
     For detailed information on setting up the Lifeline, refer to the procedure under Section 3.4.3 on page 42.

Figure 3-1: RJ-11 Telephone Connector Pinout



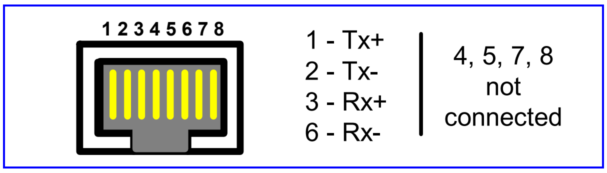
- **Digital interfaces:** Connect the E1/T1 trunk cables to the ports on the digital I/O module/s. Connect the other ends of the trunk cables to your PBX/PSTN switch. For a 1+1 or 2+2 fallback option, connect trunks 1 and 3 to your PBX, and trunks 2 and 4 to the PSTN. If the power fails, a relay connects trunks 1 to 2 and 3 to 4 (in the same module) acting as a fallback for PSTN trunks. RJ-48c trunk connectors are wired according to the figure below.

Figure 3-2: RJ-48c Trunk Connector Pinouts



3. Connect the first Ethernet connection (labeled **I**), located on the CPU module of the Mediant 1000 front panel, directly to the network using a standard RJ-45 Ethernet cable (and the second connection as optional redundancy / backup). Note that for redundant operation it is recommended to connect each of the Ethernet connectors to a different switch. For the connector's pinouts, refer to the figure below. When assigning an IP address to the Mediant 1000 using HTTP (under step 1 in Section 4.2.1), you may be required to disconnect this cable and re-cable it differently.

Figure 3-3: RJ-45 Ethernet Connector Pinout



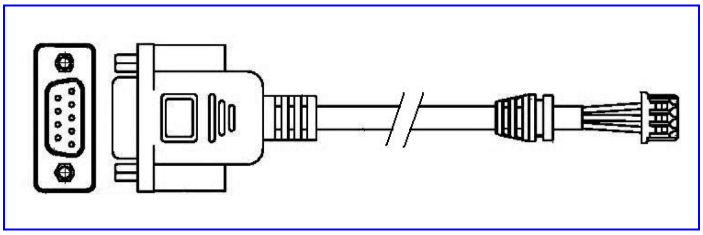
4. Optionally, connect the Mediant 1000 RS-232 port to your PC (refer to Section 3.4.1 on page 40). The RS-232 port is used to access the CLI (Section 4.2.4 on page 50) and to receive error / notification messages.
5. If you want to set up a dry contact system, refer to Section 3.4.2 on page 41.
6. Connect the left (active) 100-240V~50-60 Hz power socket to the mains using the supplied cord. For detailed information on the power supply module, refer to Section 2.1.3 on page 32.

3.4.1 Connecting the Mediant 1000 RS-232 Port to Your PC

Using the supplied cross-over RS-232 cable (shown in Figure 3-4) connect the Mediant 1000 RS-232 port (Labeled **I0I0**) to either COM1 or COM2 RS-232 communication port on your PC.

For information on establishing a serial communications link with the Mediant 1000, refer to Section 12.2 on page 319.

Figure 3-4: Supplied RS-232 Cable



3.4.2 Setting up a Dry Contact Relay Alarm System

The dry contact ports **I** and **II** located on the gateway's CPU module (refer to Figure 2-1 on page 29), allows you to connect the gateway to an external audible or visual alarm system. The table below describes the operational status of these dry contact ports.

Table 3-1: Dry Contact Operational Description

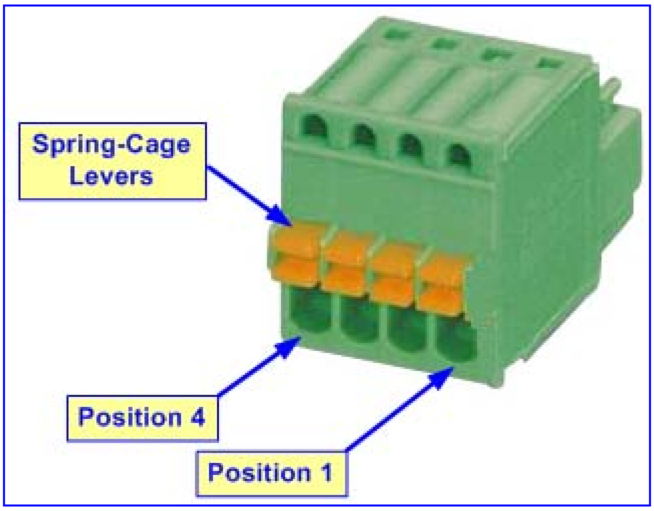
Port	Normal State	Alarm Severity State
I	During normal operation, the dry contact is open.	If a Major alarm is generated, the dry contact closes.
II	During normal operation, the dry contact is open.	If a Critical alarm is generated, the dry contact closes.

You can view a detailed description of these alarms by accessing the Active Alarms Table (refer to Section 5.10.4 on page 207) in the gateway's embedded Web server.

The external alarm system is connected to the Mediant 1000 gateway's dry contact connector on the CPU module, using the supplied dry contact wires' mate (refer to Figure 3-5). The mate provides four spring-cage terminal block connector labeled 4, 3, 2 and 1 (from left to right). These connections correspond to the four pins of the dry contact connector on the CPU module.

You need to supply your own wiring (for connecting to the mate's spring-cage connections) as well as a visual and/or audible alarm system attached at the other end of the wires. The dry contact connector suites wire sizes in the range 20 to 28 AWG. In addition, the dry contact system can receive a current of up to 1.5 A.

Figure 3-5: Dry Contact Wires' Mate



Note: The dry contact alarm provided on the CPU card should be connected only to SELV (Safety Extra-Low Voltage) non-energy hazard sources (Class 2) as per UL60950 and EN60950.

➢ To set up a dry contact system, take these 2 steps:

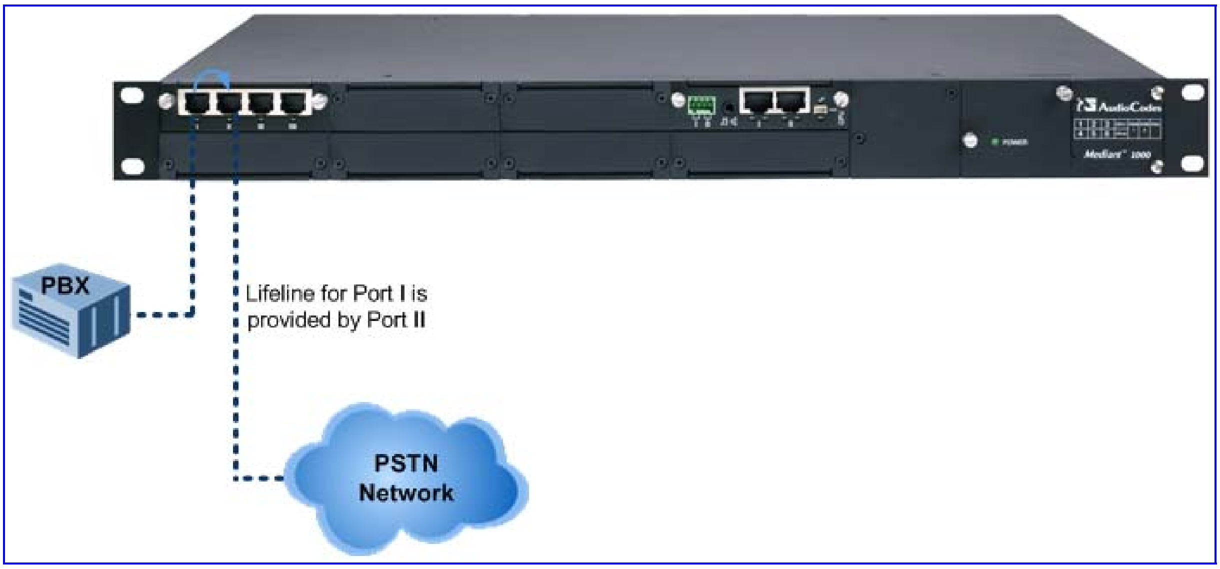
1. Insert two wires into the mate's spring-cage wire connectors in position 4 and 3 for the gateway's dry contact port **I**, and two wires in position 2 and 1 (for the gateway's dry contact port **II**), by performing the following:
   a. With a sharp, pointed object, press the position's corresponding orange button; the cage of the connection opens.
   b. Insert the wire into the connector and then release the orange button; the cage closes, securing the wire in place.
2. Connect the other ends of the dry contact wiring to the alerts system (alarm, siren, or light) according to your preferences and requirements.

3.4.3 Cabling the Digital Lifeline

The Mediant 1000 gateway containing either one or two digital modules, each with 1 or 2 pairs of spans can provide a "lifeline" telephone link. In the event of a power failure, a relay connects trunk 1 to 2, and / or 3 to 4 in the same module. The link is provided by the closing of a metallic switch inside the module so that the trunk from the PBX is routed from the module to the PSTN.

Note: The Lifeline feature can only be supported between ports on the same digital module.

Figure 3-6: Mediant 1000 Lifeline Cabling (e.g., Trunk 1 and Trunk 2)



3.4.4 Cabling the Analog Lifeline Phone

The gateway's FXS modules provide a Lifeline phone connection on Port 1.

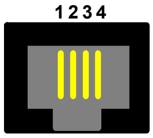
Note: Only the Mediant 1000 FXS modules support analog Lifeline.

The Lifeline provides a wired analog POTS phone connection to any PSTN or PBX FXS port when there is no power, or when the network connection fails. You can therefore, use the Lifeline phone even when the Mediant 1000 is not powered on or not connected to the network.

The Lifeline splitter connects pins 1 and 4 to another source of an FXS port, and pins 2 and 3 to the POTS phone (refer to the Lifeline pinout in Figure 3-7 on page 43).

The use of the Lifeline on network failure can be disabled using the LifeLineType *ini* file parameter (described in Table 6-2 on page 220).

Figure 3-7: Lifeline Pinout and RJ-11 Port 1 in the FXS Module



1 - Lifeline Tip
2 - Tip
3 - Ring
4 - Lifeline Ring

➢ To cable the Mediant 1000 FXS module's Lifeline, take these 3 steps:

1. Connect the Lifeline Splitter (supplied) to Port **1** on the Mediant 1000 FXS module.
2. Connect the Lifeline phone to Port **A** on the Lifeline Splitter.
3. Connect an analog PSTN line to Port **B** on the Lifeline Splitter.

Figure 3-8: Mediant 1000 Lifeline Setup

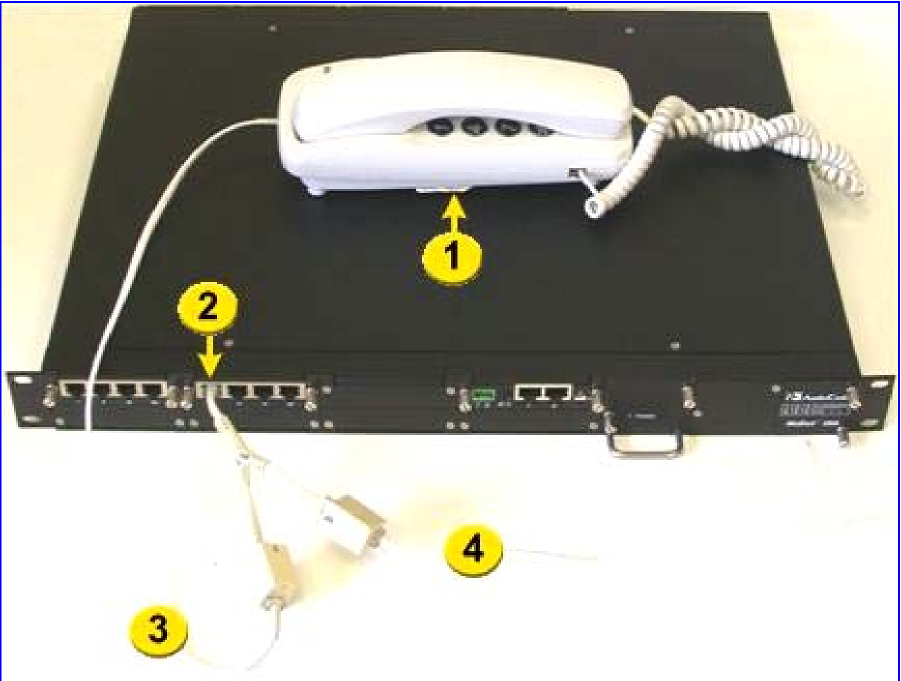


Table 3-2: Mediant 1000 Lifeline Setup Component Descriptions

Item #	Component Description
1	Lifeline phone.
2	Lifeline connected to FXS module Port **1**.
3	Splitter (**A**) to Lifeline phone.
4	Splitter (**B**) to PSTN or PBX extension analog line.

3.5 Maintenance

This section describes the following maintenance operations:

- Replacing modules (refer to Section 3.5.1 on page 44)
- Inserting additional modules (refer to Section 3.5.2 on page 44)
- Replacing the Fan Tray unit (refer to Section 3.5.3 on page 45)

3.5.1 Replacing Modules

The Mediant 1000 I/O modules are hot-swappable (except for the OSN Server modules -- refer to Appendix K on page 465). The replacement of Mediant 1000 communication modules (i.e., digital, FXS, and FXO) is performed using the Mediant 1000 embedded Web server. Once you have 'removed' the module using the Web server, you need to physical remove and then insert a new module. Once the new module is physically inserted, you then need to 'insert' it using the Web server.

Warnings:

- Replacing of a damaged module can be performed only with the same module and in the exact module slot (e.g., a module with two digital spans in Slot 1 must be replaced with a module with two digital spans in Slot 1).
- When only one module is available, removal of the module causes the device to reset.

➢ To replace Mediant 1000 modules, take these 4 steps:

1. Remove the module using the embedded Web server (refer to Section 5.10.9 on page 210).
2. Physical remove the module from the Mediant 1000 front-panel slot, by performing the following:
   a. Using a flathead screwdriver, loosen the module's two mounting screws.
   b. Gently extract the module from the slot.
3. Physical insert the new module into the same slot from where the module that you are replacing resided, by performing the following:
   a. Insert the module into the empty slot, with the plain side of the Printed Circuit Board (PCB) facing up. Make sure the PCB slides into the slot rails by aligning the module with the rails in the slot.
   b. Push the module into the slot and press on it firmly to ensure it has been fully inserted.
   c. Using a flathead screwdriver, tighten the module's mounting pins.
4. Insert the module using the embedded Web server (refer to Section 5.10.9 on page 210).

3.5.2 Inserting Modules to Previously Empty Slots

The procedure below describes how to add additional modules (i.e., digital, and FXS and FXO analog) to previously empty module slots in the gateway.

Warning: Adding a module to a previously empty slot must only be performed when the power to the gateway is switched off.

Note: The standard FXO module supports outdoor and indoor (lightning protection) loop start signaling. For ground start signaling, the **FXO G** module is required. This module supports either loop or ground start, and only supports indoor protection. (The FXS module supports both loop and ground start signaling.)

To enable ground start, use the *ini* file parameter GroundKeyDetection (refer to Section 6.5.2 on page 220).

➢ To install a module to a previously empty slot, take these 6 steps:

1. Power off the Mediant 1000.
2. On the Mediant 1000 front panel, using a Phillips screwdriver remove the black metal cover plate protecting the module slot.
3. Insert the module into the empty slot, with the plain side of the Printed Circuit Board (PCB) facing up. Make sure the PCB slides into the slot rails by aligning the module with the rails in the slot.
4. Push the module into the slot and press on it firmly to ensure it has been fully inserted.
5. Using a flathead screwdriver, tighten the module's mounting pins.
6. Power on the Mediant 1000.

3.5.3 Replacing the Air Filter

The fan tray module includes a removable air filter (located within the fan assembly, immediately inside the perforated grill). The air filter should be replaced approximately every 90 days and should be checked weekly to ensure it is not saturated and that it does not require cleaning / replacement. You should clean the air filter no more than three times, after which the air filter should be replaced.

The air filter removed from the fan tray unit is shown in Table 2-7 on page 35.

Cleaning or replacing the air filter can be carried out while the system is fully functioning.

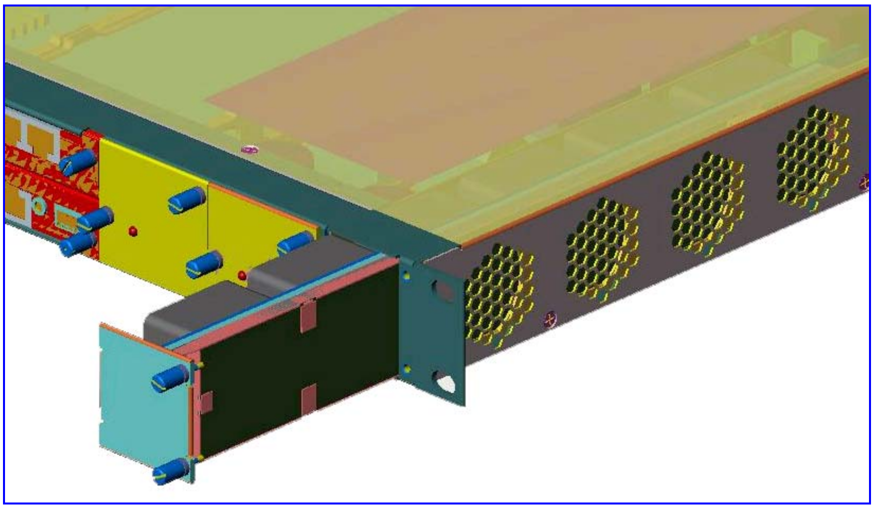
Warnings:

- When removing the fan tray unit, the fans continue rotating at a high speed. Use caution not to touch the rotating fan blades.
- Prepare all the equipment you need to clean the air filter before removing it for cleaning. It is imperative that the chassis does not remain without the fan tray unit for a long period of time. Be sure to reinsert the fan tray unit while you are cleaning the air filter and reinsert the air filter as soon as it is clean.

➢ To clean / replace the air filter, take these 7 steps:

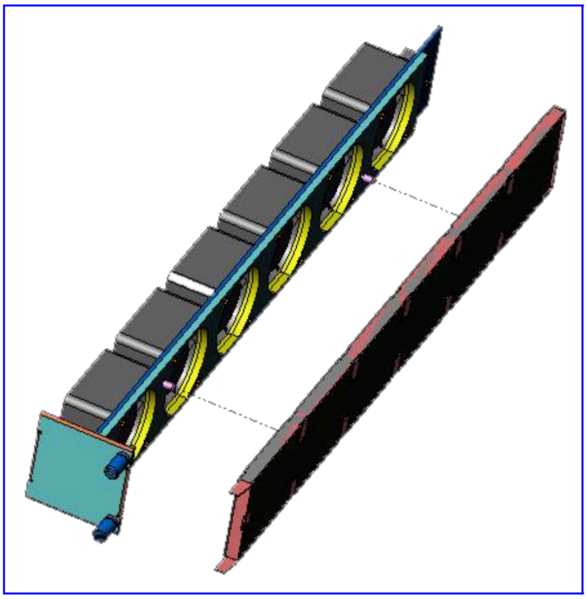
1. Release the two screws on the top right-hand corner and the bottom right-hand corner of the front panel of the fan tray unit.
2. Pull the fan tray unit outward. Figure 3-9 below shows the fan try unit slightly extracted.

Figure 3-9: Slightly Extracted Fan Try Unit



3. With your fingertips, grasp the steel frame of the air filter and separate it from the fan tray unit; you should be able to remove it relatively easily. Figure 3-10 below shows the air filter extracted from the fan tray unit.

Figure 3-10: Fan Tray with Filter Extracted



4. Take one of the following steps:
   - If you are cleaning the filter, use a vacuum cleaner (set to light suction) to remove dust particles from the filter.
   - Alternatively, if you are replacing the filter, discard the old air filter and replace it with an air filter purchased from AudioCodes.
5. Attach the (new / cleaned) air filter to the fan tray module; position the two holes on the filter over the pins on the fan tray (refer to Figure 3-10).
6. Insert the fan tray unit into its slot, until the front panel is flush with the chassis plate.
7. Fasten the two screws on the top right-hand corner and the bottom right-hand corner of the front panel of the fan tray unit.

4 Getting Started

The Mediant 1000 is supplied with default networking parameters (show in Table 4-1 below) and with an application software already resident in its flash memory (with factory default parameters).

Before you begin configuring the gateway, change its default IP address to correspond with your network environment (refer to Section 4.2) and learn about the configuration methods available on the Mediant 1000 (refer to Section 4.1 below).

For information on quickly setting up the Mediant 1000 with basic parameters using a standard Web browser, refer to Section 4.2.3 on page 49.

Table 4-1: Mediant 1000 Default Networking Parameters

Parameter	Default Value
IP Address	10.1.10.10
Subnet Mask	255.255.0.0
Default Gateway IP Address	0.0.0.0

4.1 Configuration Concepts

You can utilize the Mediant 1000 in a wide variety of applications, enabled by its parameters and configuration files (e.g., Call Progress Tones (CPT)). The parameters can be configured and configuration files can be loaded using:

- A standard Web Browser (described and explained in Section 5 on page 55).
- A configuration file referred to as the *ini* file. For information on how to use the *ini* file, refer to Section 6 on page 213.
- An SNMP browser software (refer to Section 16 on page 369).

To upgrade the Mediant 1000 (load new software or configuration files onto the gateway) use the Software Upgrade wizard, available through the Web Interface (refer to Section 5.8.1 on page 191), or alternatively use the BootP/TFTP configuration utility (refer to Section 7.3.1 on page 260).

For information on the configuration files, refer to Section 6 on page 213.

4.2 Assigning the Mediant 1000 IP Address

To assign an IP address to the Mediant 1000 use one of the following methods:

- HTTP using a Web browser (refer to Section 4.2.1 below).
- BootP (refer to Section 4.2.2 on page 48).
- Voice Menu using a standard touch-tone telephone connected to one of the FXS analog ports (refer to Section 4.2.3 on page 49). This method doesn’t apply to FXO modules.
- The embedded Command Line Interface (CLI) accessed via Telnet or RS-232 (refer to Section 4.2.4 on page 50).
- DHCP (refer to Section 7.2 on page 259).

You can use the ‘Reset’ button at any time to restore the Mediant 1000 networking parameters to their factory default values (refer to Section 12.1 on page 319).

4.2.1 Assigning an IP Address Using HTTP

➢ **To assign an IP address using HTTP, take these 8 steps:**

1. Disconnect the Mediant 1000 from the network and reconnect it to your PC using one of the following two methods:
   - Use a standard Ethernet cable to connect the network interface on your PC to a port on a network hub / switch. Use a second standard Ethernet cable to connect the Mediant 1000 to another port on the same network hub / switch.
   - Use an Ethernet cross-over cable to directly connect the network interface on your PC to the Mediant 1000.
2. Change your PC's IP address and subnet mask to correspond with the Mediant 1000 factory default IP address and subnet mask, shown in Table 4-1. For details on changing the IP address and subnet mask of your PC, refer to Windows™ Online Help (Start>Help).
3. Access the Mediant 1000 Embedded Web Server (refer to Section 5.3 on page 57).
4. In the 'Quick Setup' screen (shown in Figure 4-1 on page 52), set the Mediant 1000 'IP Address', 'Subnet Mask' and 'Default Gateway IP Address' fields under 'IP Configuration' *to correspond with your network IP settings.* If your network doesn't feature a default gateway, enter a dummy value in the 'Default Gateway IP Address' field.
5. Click the **Reset** button, and then at the prompt, click **OK**; the Medіant 1000 applies the changes and restarts.

Tip: Record and retain the IP address and subnet mask you assign the Mediant 1000. Do the same when defining new username or password. If the Embedded Web Server is unavailable (for example, if you've lost your username and password), use the BootP/TFTP (Trivial File Transfer Protocol) configuration utility to access the device, 'reflash' the load and reset the password (refer to Appendix D on page 413 for detailed information on using a BootP/TFTP configuration utility to access the device).

6. Disconnect your PC from the Mediant 1000 or from the hub / switch (depending on the connection method you used in Step 1).
7. Reconnect the Mediant 1000 and your PC (if necessary) to the LAN.
8. Restore your PC's IP address & subnet mask to what they originally were. If necessary, restart your PC and re-access the Mediant 1000 via the Embedded Web Server with its new assigned IP address.

4.2.2 Assigning an IP Address Using BootP

The below procedure describes how to assign the gateway an IP address using the supplied BootP application.

Note: BootP procedure can also be performed using any standard compatible BootP server.

Tip: You can also use BootP to load the auxiliary files to the Mediant 1000 (refer to Section 5.8.2.2 on page 198).

- **To assign an IP address using BootP, take these 3 steps:**

1. Open the BootP application (supplied with the Mediant 1000 software package).
2. Add client configuration for the Mediant 1000, refer to Section D.11.1 on page 419.
3. Use the reset button to *physically* reset the gateway causing it to use BootP; the Mediant 1000 changes its network parameters to the values provided by the BootP.

4.2.3 Assigning an IP Address Using the Voice Menu Guidance

Initial configuration of the gateway can be performed using a standard touch-tone telephone connected to one of the FXS analog ports. The voice menu can also be used to query and modify basic configuration parameters.

Note: Assigning an IP address using voice menu guidance is only possible when the Mediant 1000 houses an FXS module (analog).

- **To assign an IP address using the voice menu guidance, take these 7 steps:**

1. Connect a telephone to one of the FXS ports. Lift the handset and dial ***12345 (three stars followed by the digits 1, 2, 3, 4, 5).
2. Wait for the 'configuration menu' voice prompt to be played.
3. To change the IP address, press 1 followed by the pound key (#).
   - The current IP address of the gateway is played. Press # to change it.
   - Dial the new IP address; use the star (*) key instead of dots ("."), e.g. 192*168*0*4 and press # to finish.
   - Review the new IP address, and press 1 to save it.
4. To change the subnet mask, press 2 followed by the # key.
   - The current subnet mask of the gateway is played. Press # to change it.
   - Dial the new subnet mask; e.g. 255*255*0*0 and press # to finish.
   - Review the new subnet mask, and press 1 to save it.
5. To change the default gateway IP address, press 3 followed by the # key.
   - The current default gateway address of the gateway is played. Press # to change it.
   - Dial the new default gateway address; e.g. 192*168*0*1 and press # to finish.
   - Review the new default gateway address, and press 1 to save it.
6. Hang up the handset. Access the gateway's Embedded Web Server with the new IP address you assigned (refer to Section 5.3 on page 57).
7. Complete the gateway’s configuration and save it to non-volatile memory (refer to Section 5.9.2 on page 200).

The following configuration parameters can be queried or modified via the voice menu:

Table 4-2: Configuration Parameters Available via the Voice Menu

Item Number at Menu Prompt	Description
1	IP address
2	Subnet mask
3	Default gateway IP address
4	Primary DNS server IP address
7	DHCP enable / disable
11	MGCP call agent IP address (N/A)
12	MGCP call agent port number (N/A)
99	Voice menu password (initially 12345). **Note:** The voice menu password can also be changed using the parameter 'VoiceMenuPassword' (refer to Table 5-36 on page 176).

4.2.4 Assigning an IP Address Using the CLI

First access the CLI using a standard Telnet application or using serial communication software (e.g., HyperTerminal^{TM}) connected to the Mediant 1000 RS-232 port. Then assign the Mediant 1000 an IP address.

➢ To access the CLI via the Embedded Telnet Server, take these 3 steps:

1. To access the CLI using the embedded Telnet server, perform the following:
   a. Access the Mediant 1000 Embedded Web Server (refer to Section 5.3 on page 57).
   b. Set the parameter 'Embedded Telnet Server' (under **Advanced Configuration** > **Network Settings** > **Application Settings**) to 'Enable (Unsecured)' or 'Enable Secured (SSL)'.
   c. Click the **Maintenance** button on the main menu bar; the 'Maintenance Actions' screen is displayed.
   d. From the 'Burn to FLASH' drop-down list, select 'Yes', and then click the **Reset** button; the Mediant 1000 is shut down and re-activated. A message about the waiting period is displayed. The screen is refreshed.
   e. Use a standard Telnet application to connect to the Mediant 1000 Embedded Telnet Server. Note that if the Telnet server is set to SSL mode, a special Telnet client is required on your PC to connect to the Telnet interface over a secured connection.
   f. Login using the username ('Admin') and password ('Admin').

2. To access the CLI using the RS-232 port, perform the following:

   a. Connect the RS-232 port to your PC (refer to Section 3.4.1 on page 40).

   b. Use serial communication software (e.g., HyperTerminal™) to connect to the Mediant 1000. Set your serial communication software to the following communications port settings:

♦ Baud Rate:	115,200 bps
♦ Data bits:	8
♦ Parity:	None
♦ Stop bits:	1
♦ Flow control:	None

   The CLI prompt becomes available.

3. Assign an IP address by performing the following:

   a. At the prompt, type **`conf`**, and then press Enter; the configuration folder is accessed.

   b. To check the current network parameters, at the prompt, type **`GCP IP`**, and then press Enter; the current network settings are displayed.

   c. Change the network settings by typing:
      `SCP IP` `[ip_address] [subnet_mask] [default_gateway]`
      For example, 'SCP IP 10.13.77.7 255.255.0.0 10.13.0.1'. **Note:** This command requires you to enter all three network parameters (each separated by a space).

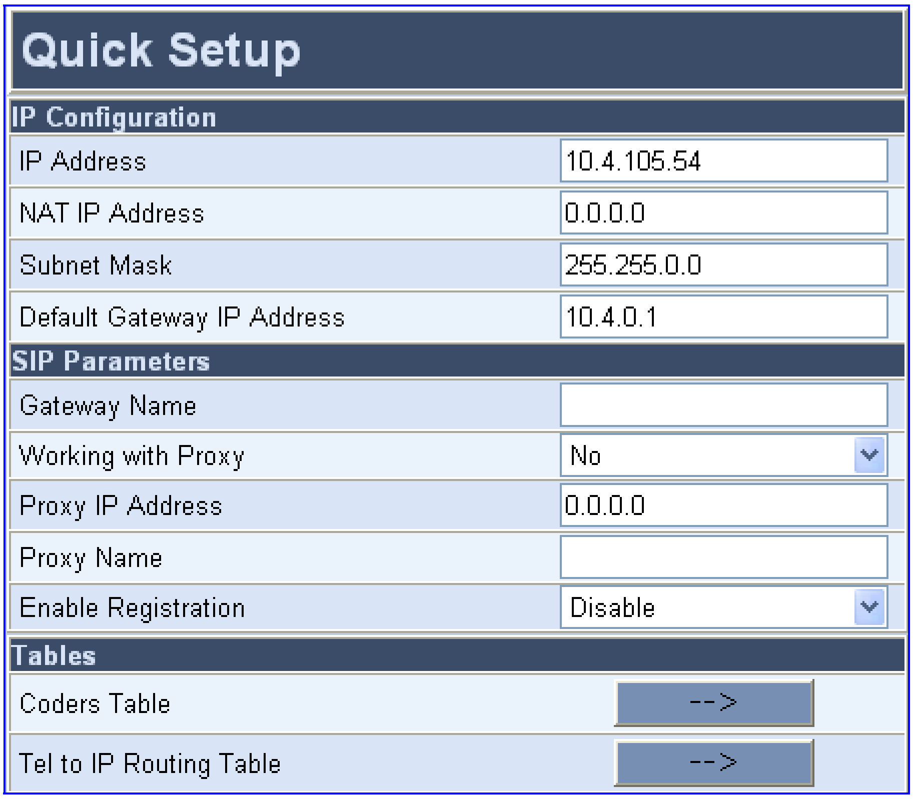
   The new settings take effect on-the-fly. Connectivity is active at the new IP address.

   d. To save the configuration, at the prompt, type **`SAR`**, and then press Enter; the Mediant 1000 restarts with the new network settings.

4.3 Configure the Mediant 1000 Basic Parameters

To configure the Mediant 1000 basic parameters use the Embedded Web Server's 'Quick Setup' screen (shown in Figure 4-1 below). Refer to Section 5.3 on page 57 for information on accessing the 'Quick Setup' screen.

Figure 4-1: Quick Setup Screen



Quick Setup

IP Configuration	
IP Address	10.4.105.54
NAT IP Address	0.0.0.0
Subnet Mask	255.255.0.0
Default Gateway IP Address	10.4.0.1
SIP Parameters	
Gateway Name	
Working with Proxy	No
Proxy IP Address	0.0.0.0
Proxy Name	
Enable Registration	Disable
Tables	
Coders Table	-->
Tel to IP Routing Table	-->

➢ To configure basic SIP parameters, take these 8 steps:

1. If the Mediant 1000 is connected to a router with Network Address Translation (NAT) enabled, perform the following procedure. If it isn't, leave the 'NAT IP Address' field undefined.
   a. Determine the 'public' IP address assigned to the router (by using, for instance, router Web management). If the public IP address is static, enter this in the 'NAT IP Address' field.
   b. Enable the DMZ (Demilitarized Zone) configuration on the router for the LAN port where the Mediant 1000 gateway is connected. This enables unknown packets to be routed to the DMZ port.
2. Under 'SIP Parameters', enter the Mediant 1000 domain name in the field 'Gateway Name'. If the field is not specified, the Mediant 1000 IP address is used instead (default).
3. When working with a Proxy server, set 'Working with Proxy' field to 'Yes', and then enter the IP address of the primary Proxy server in the field 'Proxy IP Address'. When no Proxy is used, the internal routing table is used to route the calls.
4. Enter the Proxy Name in the field 'Proxy Name'. If Proxy name is used, it replaces the Proxy IP address in all SIP messages. This means that messages are still sent to the physical Proxy IP address but the SIP URI contains the Proxy name instead.

5. Configure ‘Enable Registration’ to ‘Yes’ or ‘No’:
   - ‘No’ = the Mediant 1000 does not register to a Proxy server/Registrar (default).
   - ‘Yes’ = the Mediant 1000 registers to a Proxy server/Registrar at power up and every ‘Registration Time’ seconds; The Mediant 1000 sends a REGISTER request according to the ‘Authentication Mode’ parameter. For detailed information on the parameters ‘Registration Time’ and ‘Authentication Mode’, refer to Table 5-4 on page 72.
6. To configure the Coders Table, click the arrow button next to ‘Coders Table’. For information on how to configure the Coders Table, refer to Section 5.5.1.3 on page 78.
7. To configure the Tel to IP Routing Table, click the arrow button next to ‘Tel to IP Routing Table’. For information on how to configure the Tel to IP Routing Table, refer to Section 5.5.4.2 on page 107.
8. Click the **Reset** button, and then at the prompt, click **OK**; the Mediant 1000 applies the changes and restarts.

You are now ready to start using the VoIP gateway. To prevent unauthorized access to the Mediant 1000, it is recommended that you change the username and password that are used to access the Web Interface. Refer to Section 5.6.7.1 on page 170 for details on how to change the username and password.

Note: The preferred coder is the coder that the Mediant 1000 uses as a first choice for all connections. If the far end gateway does not use this coder, the Mediant 1000 negotiates with the far end gateway to select a coder that both sides can use.

Tip: Once the gateway is configured correctly back up your settings by making a copy of the VoIP gateway configuration (*ini* file) and store it in a directory on your PC. This saved file can be used to restore configuration settings at a future time. For information on backing up and restoring the gateway’s configuration, refer to Section 5.6.5 on page 168.

Reader's Notes

5 Web Management

The Embedded Web Server is used both for gateway configuration, including loading of configuration files, and for run-time monitoring. The Embedded Web Server can be accessed from a standard Web browser, such as Microsoft™ Internet Explorer, Netscape™ Navigator, etc. Specifically, users can employ this facility to set up the gateway configuration parameters. Users also have the option to remotely reset the gateway and to permanently apply the new set of parameters.

5.1 Computer Requirements

To use the Embedded Web Server, the following is required:

- A computer capable of running your Web browser.
- A network connection to the VoIP gateway.
- One of the following compatible Web browsers:
    - Microsoft™ Internet Explorer™ (version 6.0 and higher).
    - Netscape™ Navigator™ (version 7.2 and higher).

Note: The browser must be Java-script enabled. If java-script is disabled, access to the Embedded Web Server is denied.

5.2 Protection and Security Mechanisms

Access to the Embedded Web Server is controlled by the following protection and security mechanisms:

- User accounts (refer to Section 5.2.1 below).
- Read-only mode (refer to Section 5.2.2).
- Disabling access (refer to Section 5.2.3).
- Limiting access to a predefined list of IP addresses (refer to Section 5.6.7.2 on page 172).
- Secured HTTP connection (HTTPS) (refer to Section 14.2.2 on page 352).
- Managed access using a RADIUS server (refer to Section 14.4 on page 357).

5.2.1 User Accounts

To prevent unauthorized access to the Embedded Web Server, two user accounts are available: primary and secondary. Each account is composed of three attributes: username, password and access level. The username and password enable access to the Embedded Web Server itself; the access level determines the extent of the access (i.e., availability of screens and read / write privileges). Note that additional accounts can be defined using a RADIUS server (refer to Section 14.4 on page 357).

Table 5-1 lists the available access levels and their privileges.

Table 5-1: Available Access Levels and their Privileges

Access Level	Numeric Representation*	Privileges
Security Administrator	200	Read / write privileges for all screens
Administrator	100	Read-only privilege for security-related screens and read / write privileges for the others
User Monitor	50	No access to security-related and file-loading screens and read-only access to the others
No Access	0	No access to any screen

* The numeric representation of the access level is used only to define accounts in a RADIUS server (the access level ranges from 1 to 255).

The access level mechanism operation is as follows (for both Web and RADIUS accounts):

Each Web screen features two (hard-coded) minimum access levels, read and write. The read access level determines whether the screen can be viewed. The write access level determines whether the information in the screen can be modified.

When a user tries to access a specific Web screen, the user's access level is compared with the access levels of the screen:

- If the access level of the user is less than the screen's read access level, the screen cannot be viewed.
- If the access level of the user is equal to or greater than the screen's read access level but less than the write access level, the screen is read only.
- If the access level of the user is equal to or greater than the screen's write access level, the screen can be modified.

The default attributes for the two accounts are shown in Table 5-2 below:

Table 5-2: Default Attributes for the Accounts

Account / Attribute	Username (Case-Sensitive)	Password (Case-Sensitive)	Access Level
Primary Account	Admin	Admin	Security Administrator*
Secondary Account	User	User	User Monitor

* The access level of the primary account cannot be changed; all other account-attributes can be modified.

The first time a browser request is made, users are requested to provide their account's username and password to obtain access. If the Embedded Web Server is left idle for more than five minutes, the session expires and the user is required to re-enter username and password.

Tip: To access the Embedded Web Server with a different account, click the **Log Off** button and re-access with a new username and password.

For details on changing the account attributes, refer to Section 5.6.7.1 on page 170. Note that the password and username can be a maximum of 19 case-sensitive characters.

To reset the username and password of both accounts to their defaults, set the *ini* file parameter 'ResetWebPassword' to 1.

5.2.2 Limiting the Embedded Web Server to Read-Only Mode

Users can limit access to the Embedded Web Server to read-only mode by changing the *ini* file parameter 'DisableWebConfig' to 1. In this mode all Web screens, regardless of the access level used, are read-only and cannot be modified. In addition, the following screens cannot be accessed: 'Quick Setup', 'Web User Accounts', 'Maintenance Actions' and all file-loading screens.

Note 1: Read-only policy can also be applied to selected users by setting the access level of the secondary account to 'User Monitor' (DisableWebConfig = 0) and distributing the primary and secondary accounts to users according to the organization's security policy.

Note 2: When DisableWebConfig is set to 1, read-only privileges are applied to all accounts regardless of their access level.

5.2.3 Disabling the Embedded Web Server

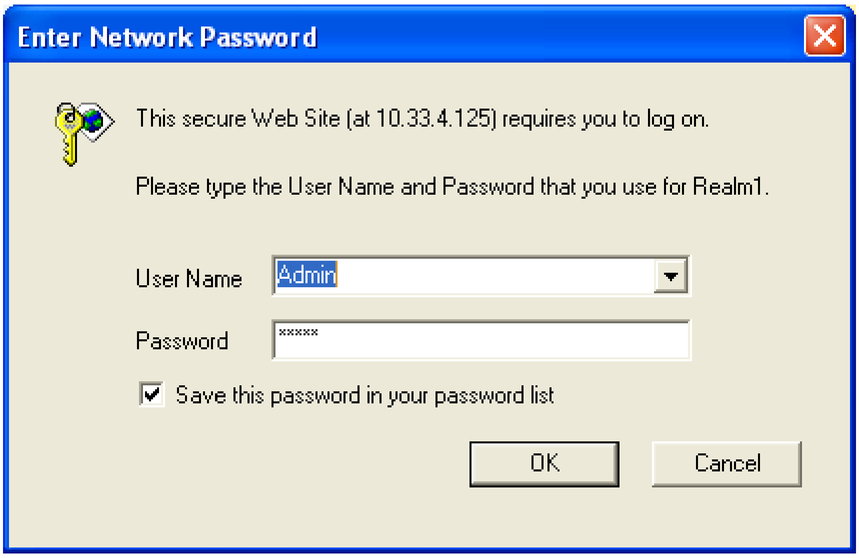
Access to the Embedded Web Server can be disabled by using the *ini* file parameter 'DisableWebTask = 1'. The default is access enabled.

5.3 Accessing the Embedded Web Server

➢ To access the Embedded Web Server, take these 4 steps:

1. Open a standard Web-browsing application such as Microsoft™ Internet Explorer™ or Netscape™ Navigator™.
2. In the Uniform Resource Locator (URL) field, specify the IP address of the Mediant 1000 (e.g., http://10.1.10.10); the Embedded Web Server's 'Enter Network Password' screen appears, shown in Figure 5-1.

Figure 5-1: Embedded Web Server Login Screen



Enter Network Password

This secure Web Site (at 10.33.4.125) requires you to log on.

Please type the User Name and Password that you use for Realm1.

User Name: Admin

Password: *****

☑ Save this password in your password list

3. In the 'User Name' and 'Password' fields, enter the username (default: 'Admin') and password (default: 'Admin'). Note that the username and password are case-sensitive.
4. Click the **OK** button; the 'Quick Setup' screen is accessed (shown in Figure 4-1 on page 52).

5.3.1 Using Internet Explorer to Access the Embedded Web Server

Internet explorer's security settings may block access to the gateway's Web browser if they're configured incorrectly. In this case, the following message is displayed:

Unauthorized

Correct authorization is required for this area. Either your browser does not perform authorization or your authorization has failed. RomPager server.

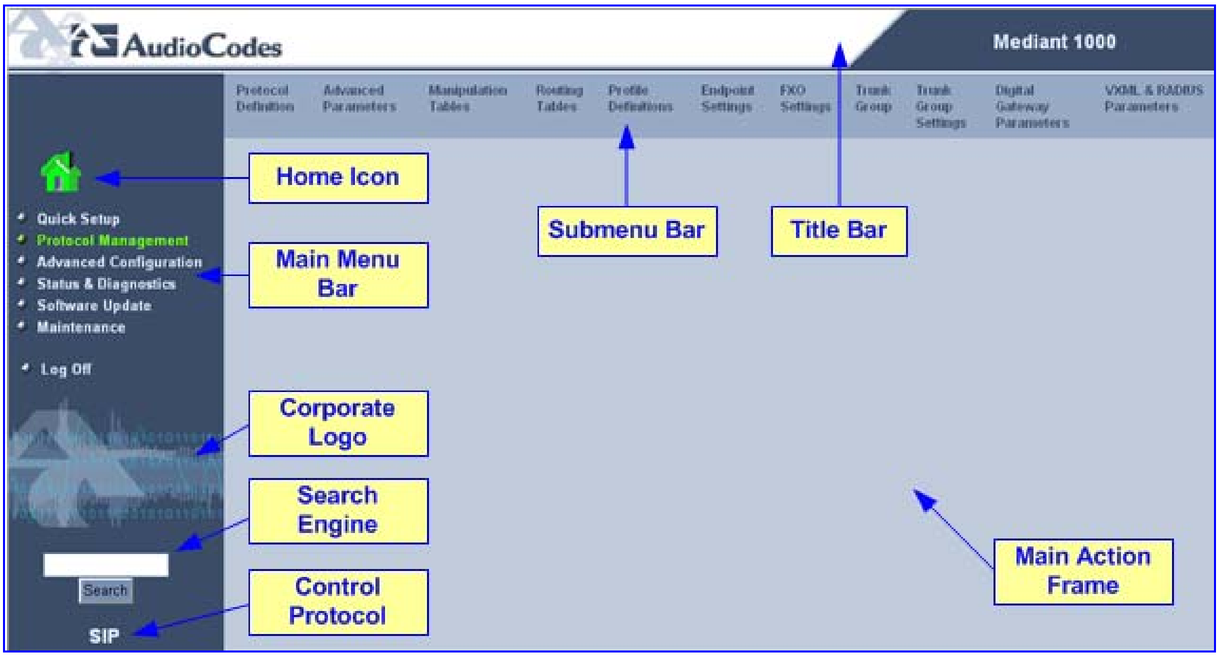
➢ To troubleshoot blocked access to Internet Explorer™, take these 3 steps

1. Delete all cookies from the Temporary Internet files. If this does not clear up the problem, the security settings may need to be altered (continue with Step 2).
2. In Internet Explorer, Tools, Internet Options:
   - Select the **Security** tab, select Custom Level. Scroll down until the Logon options are displayed and change the setting to: **Prompt for username and password**.
   - Select the Advanced tab, scroll down until the HTTP 1.1 Settings are displayed and verify that the **Use HTTP 1.1** option is checked.
3. Restart the browser.

5.4 Getting Acquainted with the Web Interface

Figure 5-2 shows the general layout of the Web Interface screen.

Figure 5-2: Mediant 1000 Web Interface



The Web Interface screen features the following components:

- **Title bar:** contains three configurable elements: corporate logo, a background image and the product's name. For information on how to modify these elements, refer to Section 12.5 on page 325.
- **Product name:** the gateway's name.
- **Main menu bar:** always appears on the left of every screen to quickly access parameters, submenus, submenu options, functions and operations.

- **Submenu bar:** appears on the top of screens and contains submenu options.
- **Main action frame:** the main area of the screen in which information is viewed and configured.
- **Home icon:** when clicked it opens the Home page screen (refer to Section 5.10 on page 202).
- **Corporate logo:** AudioCodes' corporate logo. For information on how to remove this logo Section 12.5 on page 325.
- **Search engine:** for searching *ini* file parameters that are configured using the Embedded Web Server (refer to Section 5.4.4 on page 60).
- **Control Protocol:** the Mediant 1000 control protocol.

5.4.1 Main Menu Bar

The main menu bar of the Web Interface is divided into the following menus:

- **Quick Setup:** Use this menu to configure the gateway's basic settings; for the full list of configurable parameters go directly to 'Protocol Management' and 'Advanced Configuration' menus. An example of the Quick Setup configuration is described in Section 4.2.3 on page 49.
- **Protocol Management:** Use this menu to configure the gateway's control protocol parameters and tables (refer to Section 5.4.4 on page 60).
- **Advanced Configuration:** Use this menu to set the gateway's advanced configuration parameters (for advanced users only) (refer to Section 5.6 on page 138).
- **Status & Diagnostics:** Use this menu to view Syslog messages, hardware / software product information, and to assess the gateway's statistics and IP connectivity information (refer to Section 5.7 on page 184).
- **Software Update:** Use this menu when you want to load new software or configuration files onto the gateway (refer to Section 5.8 on page 191).
- **Maintenance:** Use this menu to remotely lock/unlock the device (refer to Section 5.9.1 on page 198), save configuration changes to the non-volatile flash memory (refer to Section 5.9.2 on page 200), and reset the gateway (refer to Section 5.9.3 on page 200).

When positioning your curser over a parameter name (or a table) for more than 1 second, a short description of this parameter is displayed. Note that parameters preceded by an exclamation mark (!) are *not* changeable on-the-fly and require that the device be reset.

5.4.2 Saving Changes

To save changes to the volatile memory (RAM) click the **Submit** button (changes to parameters with on-the-fly capabilities are immediately available, other parameter are updated only after a gateway reset). Parameters that are only saved to the volatile memory revert to their previous settings after hardware reset. When performing a software reset (i.e., via Web or SNMP) you can choose to save the changes to the non-volatile memory. To save changes so they are available after a power fail, you must save the changes to the non-volatile memory (flash).

To save the changes to flash, refer to Section 5.9.2 on page 200.

5.4.3 Entering Phone Numbers in Various Tables

Phone numbers entered into various tables on the gateway, such as the Tel to IP routing table, must be entered without any formatting characters. For example, if you wish to enter the phone number 555-1212, it must be entered as 5551212 without the hyphen (-). If the hyphen is entered, the entry does not work. The hyphen character is used in number entry only, as part of a range definition. For example, the entry [20-29] means 'all numbers in the range 20 to 29'.

5.4.4 Searching for Configuration Parameters

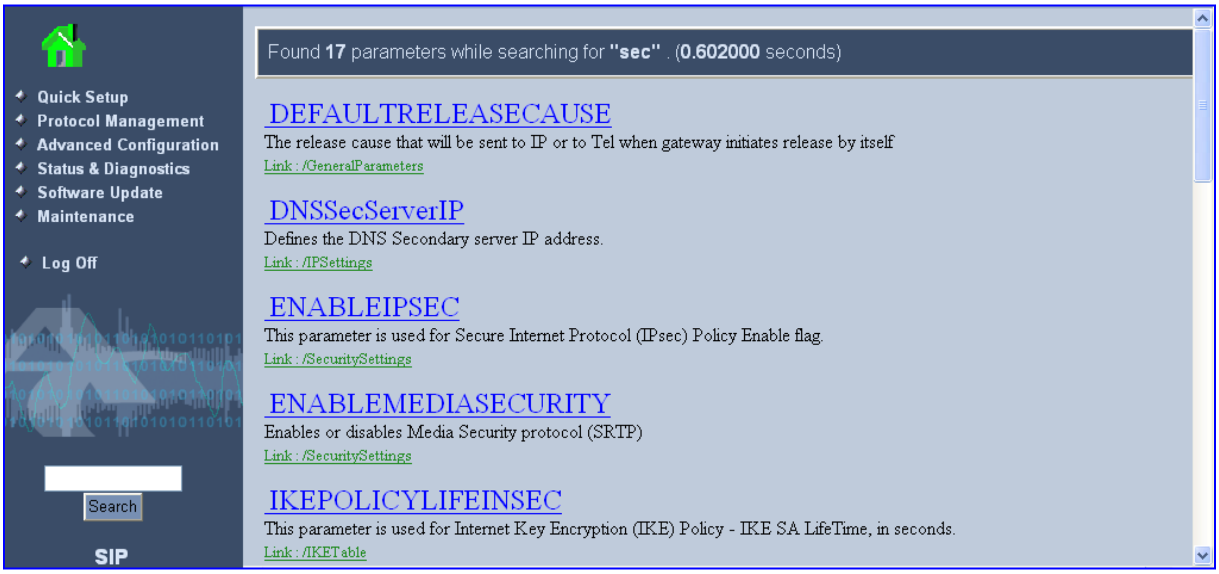
The Embedded Web Server provides a search engine that allows you to search any *ini* file parameter that is configurable by the Web server. The **Search** button, located near the bottom of the Main menu bar is used to perform parameter searches.

You can search for a specific parameter (e.g., "EnableIPSec") or a sub-string of that parameter (e.g., "sec"). If you search for a sub-string, the Embedded Web Server lists all parameters that contain the searched sub-string in their parameter names.

➢ To search for an ini file parameter configurable by the Web server, take these 3 steps:

1. In the Search Engine field, enter the parameter name or sub-string of the parameter name.
2. Click **Search**. The Searched Result screen appears, listing all searched parameter results.

Figure 5-3: Searched Result Screen



Found **17** parameters while searching for **"sec"** . (**0.602000** seconds)

DEFAULTRELEASECAUSE
The release cause that will be sent to IP or to Tel when gateway initiates release by itself
Link : /GeneralParameters

DNSSecServerIP
Defines the DNS Secondary server IP address.
Link : /IPSettings

ENABLEIPSEC
This parameter is used for Secure Internet Protocol (IPsec) Policy Enable flag.
Link : /SecuritySettings

ENABLEMEDIASECURITY
Enables or disables Media Security protocol (SRTP)
Link : /SecuritySettings

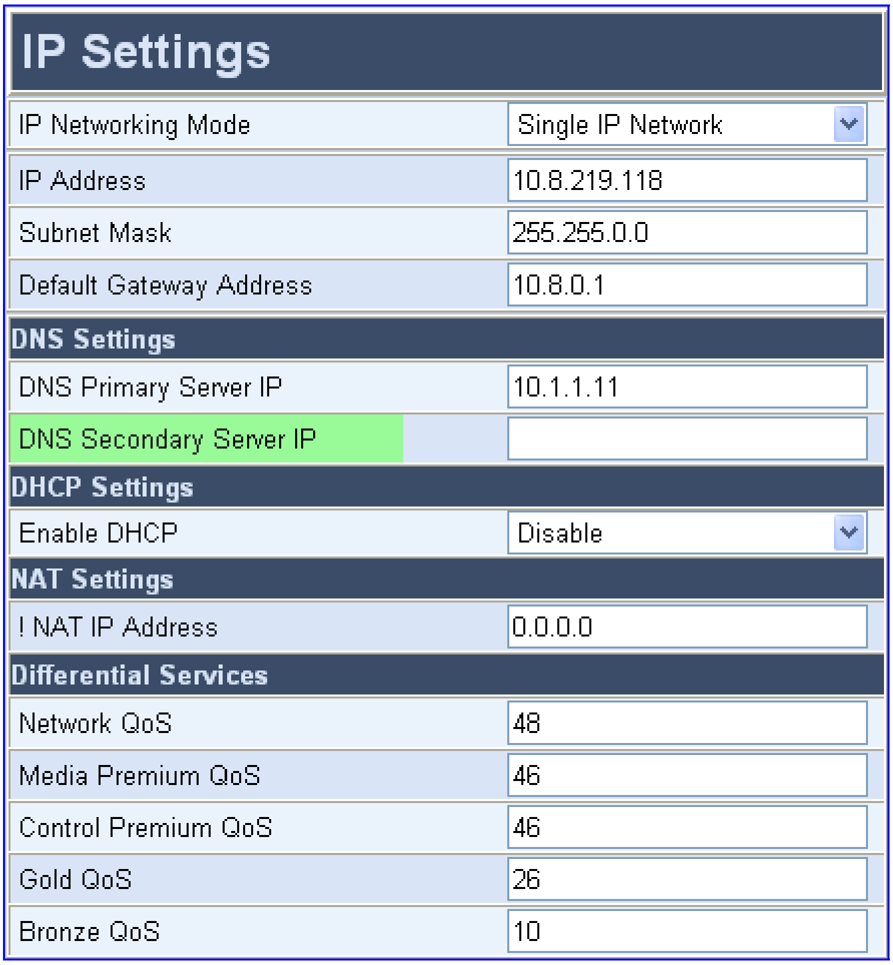
IKEPOLICYLIFEINSEC
This parameter is used for Internet Key Encryption (IKE) Policy - IKE SA LifeTime, in seconds.
Link : /IKETable

Each searched result displays the following:

- Parameter name (hyperlinked to its location in the Embedded Web Server)
- Brief description of the parameter
- Hyperlink in green displaying the URL path to its location in the Embedded Web Server location

3. In the searched result list, click the required parameter to open the screen in which the parameter appears. The relevant parameter is highlighted in green in the screen for easy viewing.

Figure 5-4: Searched Parameter Highlighted in Screen



IP Settings

IP Networking Mode	Single IP Network
IP Address	10.8.219.118
Subnet Mask	255.255.0.0
Default Gateway Address	10.8.0.1
DNS Settings	
DNS Primary Server IP	10.1.1.11
DNS Secondary Server IP	
DHCP Settings	
Enable DHCP	Disable
NAT Settings	
! NAT IP Address	0.0.0.0
Differential Services	
Network QoS	48
Media Premium QoS	46
Control Premium QoS	46
Gold QoS	26
Bronze QoS	10

Note: If the searched parameter is not located, the "No Matches Found For This String" message is displayed.

5.5 Protocol Management

The Protocol Management menu is used to configure the gateway's SIP parameters and tables.

Note: Throughout this chapter, parameters contained within square brackets are the names used to configure the parameters via the *ini* file.

5.5.1 Protocol Definition Parameters

Use this submenu to configure the following gateway's specific SIP protocol parameters:

- General Parameters (refer to Section 5.5.1.1 on page 62)
- Proxy & Registration (refer to Section 5.5.1.2 on page 70)
- Coders (refer to Section 5.5.1.3 on page 78)
- DTMF & Dialing (refer to Section 5.5.1.4 on page 80)

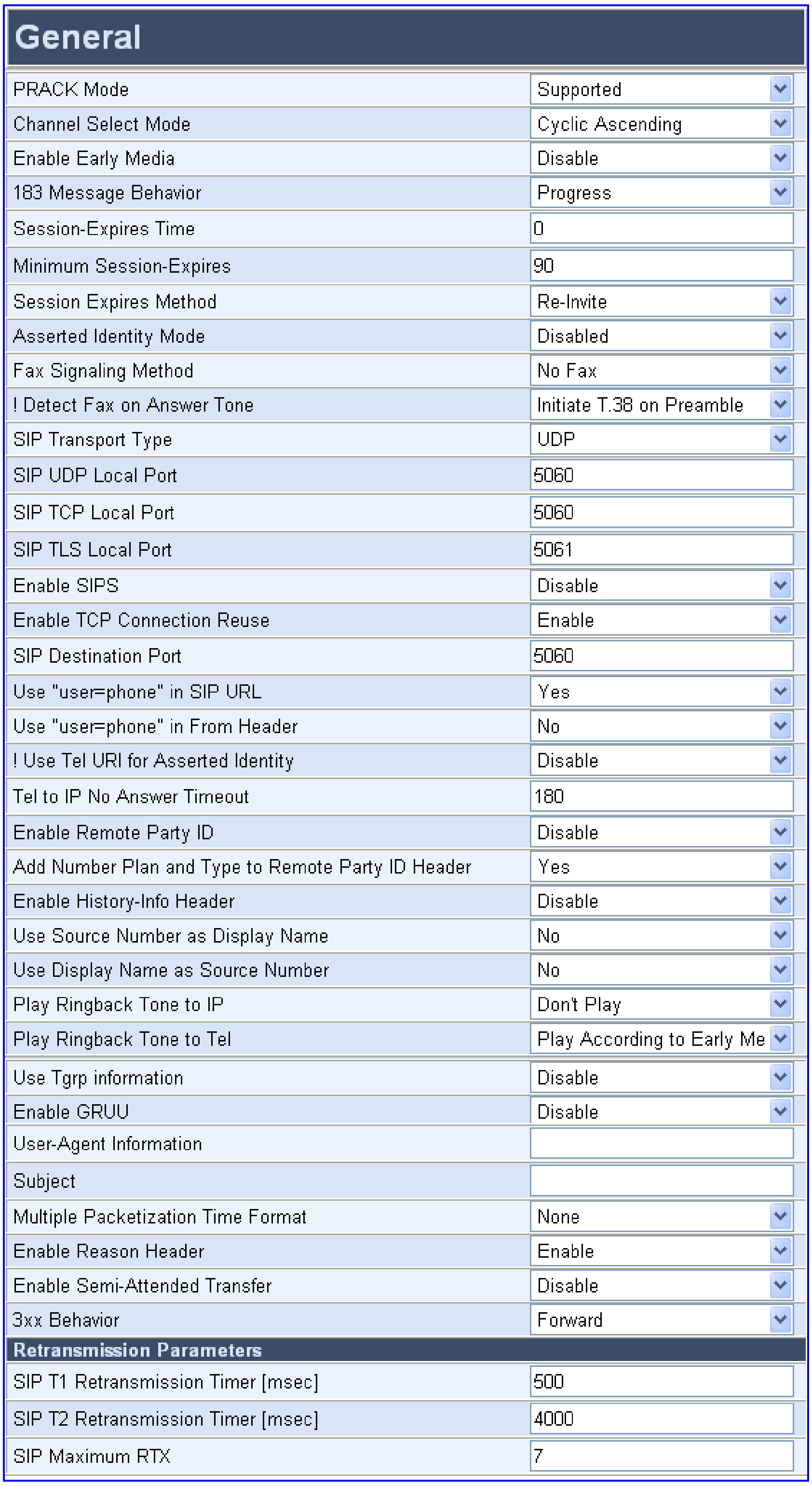
5.5.1.1 General Parameters

The General Parameters option is used to configure general SIP parameters.

➢ **To configure the general parameters under Protocol Definition, take these 4 steps:**

1. Open the 'General Parameters' screen (**Protocol Management** menu > **Protocol Definition** submenu > **General Parameters** option); the 'General Parameters' screen is displayed.

Figure 5-5: Protocol Definition, General Parameters Screen



General

PRACK Mode	Supported
Channel Select Mode	Cyclic Ascending
Enable Early Media	Disable
183 Message Behavior	Progress
Session-Expires Time	0
Minimum Session-Expires	90
Session Expires Method	Re-Invite
Asserted Identity Mode	Disabled
Fax Signaling Method	No Fax
! Detect Fax on Answer Tone	Initiate T.38 on Preamble
SIP Transport Type	UDP
SIP UDP Local Port	5060
SIP TCP Local Port	5060
SIP TLS Local Port	5061
Enable SIPS	Disable
Enable TCP Connection Reuse	Enable
SIP Destination Port	5060
Use "user=phone" in SIP URL	Yes
Use "user=phone" in From Header	No
! Use Tel URI for Asserted Identity	Disable
Tel to IP No Answer Timeout	180
Enable Remote Party ID	Disable
Add Number Plan and Type to Remote Party ID Header	Yes
Enable History-Info Header	Disable
Use Source Number as Display Name	No
Use Display Name as Source Number	No
Play Ringback Tone to IP	Don't Play
Play Ringback Tone to Tel	Play According to Early Me
Use Tgrp information	Disable
Enable GRUU	Disable
User-Agent Information	
Subject	
Multiple Packetization Time Format	None
Enable Reason Header	Enable
Enable Semi-Attended Transfer	Disable
3xx Behavior	Forward
Retransmission Parameters	
SIP T1 Retransmission Timer [msec]	500
SIP T2 Retransmission Timer [msec]	4000
SIP Maximum RTX	7

2. Configure the general parameters according to Table 5-3.

3. Click the **Submit** button to save your changes.

4. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Table 5-3: Protocol Definition, General Parameters (continues on pages 64 to 70)

Parameter	Description
PRACK Mode **[PRACKMode]**	PRACK mechanism mode for 1xx reliable responses: Disable **[0]**. Supported **[1]** (default). Required **[2]**. **Note 1:** The Supported and Required headers contain the '100rel' parameter. **Note 2:** Mediant 1000 sends PRACK message if 180/183 response is received with '100rel' in the Supported or the Required headers.
Channel Select Mode **[ChannelSelectMode]**	Port allocation algorithm for IP to Tel calls. You can select one of the following methods: ▪ By Dest Phone Number **[0]** = (default) Select the gateway port according to the called number (called number is defined in the 'Endpoint Phone Number' table). ▪ Cyclic Ascending **[1]** = Select the next available channel in an ascending cycle order. Always select the next higher channel number in the trunk group. When the gateway reaches the highest channel number in the trunk group, it selects the lowest channel number in the trunk group and then starts ascending again. ▪ Ascending **[2]** = Select the lowest available channel. Always start at the lowest channel number in the trunk group and if that channel is not available, select the next higher channel. ▪ Cyclic Descending **[3]** = Select the next available channel in descending cycle order. Always select the next lower channel number in the hunt group. When the gateway reaches the lowest channel number in the hunt group, it selects the highest channel number in the hunt group and then starts descending again. ▪ Descending **[4]** = Select the highest available channel. Always start at the highest channel number in the hunt group and if that channel is not available, select the next lower channel. ▪ Dest Number + Cyclic Ascending **[5]** = First select the gateway port according to the called number. If the called number isn't found, then select the next available channel in ascending cyclic order. Note that if the called number is found, but the port associated with this number is busy, the call is released. ▪ By Source Phone Number **[6]** = Select the gateway port according to the calling number (refer to the note below). **Note:** The internal numbers of the gateway's B-channels are defined by the TrunkGroup_x parameter (under 'Phone Number').
Enable Early Media **[EnableEarlyMedia]**	No **[0]** = Early Media is disabled (default). Yes **[1]** = Enable Early Media. If enabled, the gateway sends 183 Session Progress response with SDP (instead of 180 Ringing), allowing the media stream to be set up prior to the answering of the call. ▪ **Analog:** Note that to send 183 response you must also set the parameter 'ProgressIndicator2IP' to 1. If it is equal to 0, 180 Ringing response is sent. ▪ **Digital:** Sending 183 response depends on the Progress Indicator. It is sent only if PI = 1 or PI = 8 was received in Proceeding or Alert PRI messages. For CAS gateways see the 'ProgressIndicator2IP' parameter.
183 Message Behavior **[SIP183Behavior]**	Defines the ISDN message that is sent when 183 Session Progress message is received for IP-to-Tel calls. **[0]** = Progress message (default). **[1]** = Alert message. When set to 1, the gateway sends an Alert message (after the receipt of a 183 response) instead of an ISDN Progress message.
Session-Expires Time **[SIPSessionExpires]**	Determines the timeout (in seconds) for keeping a re-INVITE message alive within a SIP session. The SIP session is refreshed each time this timer expires. The SIP method used for session-timer updates is determined according to the parameter SessionExpiresMethod. The default is 0 (not activated).

Table 5-3: Protocol Definition, General Parameters (continues on pages 64 to 70)

Parameter	**Description**
Minimum Session-Expires **[MINSE]**	Defines the time (in seconds) that is used in the Min-SE header. This header defines the minimum time that the user agent supports for session refresh. The valid range is 10 to 100000. The default value is 90.
Session Expires Method **[SessionExpiresMethod]**	Defines the SIP method used for session-timer updates. Re-INVITE **[0]** = Use Re-INVITE messages for session-timer updates (default). UPDATE **[1]** = Use UPDATE messages. **Note 1:** The gateway can receive session-timer refreshes using both methods. **Note 2:** The UPDATE message used for session-timer purposes is not included in the SDP body.
Asserted Identity Mode **[AssertedIdMode]**	Disable **[0]** = None (default). Adding PAsserted Identity **[1]**. Adding PPreferred Identity **[2]**. The Asserted ID mode defines the header that is used in the generated INVITE request. The header also depends on the calling Privacy: allowed or restricted. The P-asserted (or P-preferred) headers are used to present the originating party's Caller ID. The Caller ID is composed of a Calling Number and (optionally) a Calling Name. P-asserted (or P-preferred) headers are used together with the Privacy header. If Caller ID is restricted, the 'Privacy: id' is included. Otherwise for allowed Caller ID the 'Privacy: none' is used. If Caller ID is restricted (received from Tel or configured in the gateway), the From header is set to <anonymous@anonymous.invalid>.
Fax Signaling Method **[IsFaxUsed]**	Determines the SIP signaling method used to establish and convey a fax session after a fax is detected. No Fax **[0]** = No fax negotiation using SIP signaling (default). T.38 Relay **[1]** = Initiates T.38 fax relay. G.711 Transport **[2]** = Initiates fax using the coder G.711 A-law/μ-law with adaptations (refer to note 1). Fax Fallback **[3]** = Initiates T.38 fax relay. If the T.38 negotiation fails, the gateway re-initiates a fax session using the coder G.711 A-law/μ-law with adaptations (see note 1). **Note 1:** Fax adaptations: Echo Canceller = On Silence Compression = Off Echo Canceller Non-Linear Processor Mode = Off Dynamic Jitter Buffer Minimum Delay = 40 Dynamic Jitter Buffer Optimization Factor = 13 **Note 2:** If the gateway initiates a fax session using G.711 (option 2 and possibly 3), a 'gpmd' attribute is added to the SDP in the following format: For A-law: 'a=gpmd:0 vbd=yes;ecan=on'. For μ-law: 'a=gpmd:8 vbd=yes;ecan=on'. **Note 3:** When 'IsFaxUsed' is set to 1, 2 or 3 the parameter 'FaxTransportMode' is ignored. **Note 4:** When the value of IsFaxUsed is other than 1, T.38 might still be used without the control protocol's involvement. To completely disable T.38, set FaxTransportMode to a value other than 1.
Detect Fax on Answer Tone **[DetFaxOnAnswerTone]**	Initiate T.38 on Preamble **[0]** = Terminating fax gateway initiates T.38 session on receiving of HDLC preamble signal from fax (default) Initiate T.38 on CED **[1]** = Terminating fax gateway initiates T.38 session on receiving of CED answer tone from fax. **Note:** This parameters is applicable only if 'IsFaxUsed = 1'.

Table 5-3: Protocol Definition, General Parameters (continues on pages 64 to 70)

Parameter	Description
SIP Transport Type **[SIPTransportType]**	Determines the *default* transport layer used for outgoing SIP calls initiated by the gateway. UDP **[0]** (default). TCP **[1]**. TLS **[2]** (SIPS). **Note:** It is recommended to use TLS to communicate with a SIP Proxy and not for direct gateway-gateway communication.
SIP UDP Local Port **[LocalSIPPort]**	Local UDP port used to receive SIP messages. The valid range is 1 to 65534. The default value is 5060.
SIP TCP Local Port **[TCPLocalSIPPort]**	Local TCP port used to receive SIP messages. The default value is 5060.
SIP TLS Local Port **[TLSLocalSIPPort]**	Local TLS port used to receive SIP messages. The default value is 5061. **Note:** The value of 'TLSLocalSIPPort' must be different to the value of 'TCPLocalSIPPort'.
Enable SIPS **[EnableSIPS]**	Enables secured SIP (SIPS) connections over multiple hops. Disable **[0]** (default). Enable **[1]**. When SIPTransportType = 2 (TLS) and EnableSIPS is disabled, TLS is used for the next network hop only. When SIPTransportType = 2 (TLS) or 1 (TCP) and EnableSIPS is enabled, TLS is used through the entire connection (over multiple hops). **Note:** If SIPS is enabled and SIPTransportType = UDP, the connection fails.
Enable TCP Connection Reuse **[EnableTCPConnectionReuse]**	Enables the reuse of the same TCP connection for all calls to the same destination. Valid options include: 0 = Use a separate TCP connection for each call (default) 1 = Use the same TCP connection for all calls
SIP Destination Port **[SIPDestinationPort]**	SIP destination port for sending initial SIP requests. The valid range is 1 to 65534. The default port is 5060. **Note:** SIP responses are sent to the port specified in the Via header.
Use "user=phone" in SIP URL **[IsUserPhone]**	No **[0]** = 'user=phone' string isn't used in SIP URI. Yes **[1]** = 'user=phone' string is part of the SIP URI (default).
Use "user=phone" in From Header **[IsUserPhoneInFrom]**	No **[0]** = Doesn't use ';user=phone' string in From header (default). Yes **[1]** = ';user=phone' string is part of the From header.
Use Tel URI for Asserted Identity **[UseTelURIForAssertedID]**	Determines the format of the URI in the P-Asserted and P-Preferred headers. 0 = 'sip:' (default). 1 = 'tel:'.
Tel to IP No Answer Timeout **[IPAlertTimeout]**	Defines the time (in seconds) the gateway waits for a 200 OK response from the called party (IP side) after sending an INVITE message. If the timer expires, the call is released. The valid range is 0 to 3600. The default value is 180.
Enable Remote Party ID **[EnableRPIheader]**	Enable Remote-Party-ID (RPI) headers for calling and called numbers for Tel→IP calls. Disable **[0]** (default). Enable **[1]** = RPI headers are generated in SIP INVITE messages for both called and calling numbers.
Add Number Plan and Type to Remote Party ID Header **[AddTON2RPI]**	No **[0]** = TON/PLAN parameters aren't included in the RPID header. Yes **[1]** = TON/PLAN parameters are included in the RPID header (default). If RPID header is enabled (EnableRPIHeader = 1) and 'AddTON2RPI=1', it is possible to configure the calling and called number type and number plan using the Number Manipulation tables for Tel→IP calls.

Table 5-3: Protocol Definition, General Parameters (continues on pages 64 to 70)

<table>
<tr><th>Parameter</th><th>Description</th></tr>
<tr><td><b>EnableHistoryInfo</b>
[Enable History-Info Header]</td><td>Enables usage of the History-Info header.
Valid options include:
0 = Disable (default)
1 = Enable
UAC Behavior:
▪ Initial request: The History-Info header is equal to the Request URI. If a PSTN Redirect number is received, it is added as an additional History-Info header with an appropriate reason.
▪ Upon receiving the final failure response, the gateway copies the History-Info as is, adds the reason of the failure response to the last entry, and concatenates a new destination to it (if an additional request is sent).
The order of the reasons is as follows:
- Q.850 Reason
- SIP Reason
- SIP Response code
▪ Upon receiving the final (success or failure) response, the gateway searches for a Redirect reason in the History-Info (i.e., 3xx/4xx SIP Reason). If found, it is passed to ISDN, according to the following table:
<table>
<tr><td>SIP Reason Code</td><td>ISDN Redirecting Reason</td></tr>
<tr><td>302 – Moved Temporarily</td><td>Call Forward Universal (CFU)</td></tr>
<tr><td>408 – Request Timeout</td><td rowspan="3">Call Forward No Answer (CFNA)</td></tr>
<tr><td>480 – Temporarily Unavailable</td></tr>
<tr><td>487 – Request Terminated</td></tr>
<tr><td>486 – Busy Here</td><td rowspan="2">Call Forward Busy (CFB)</td></tr>
<tr><td>600 – Busy Everywhere</td></tr>
</table>
▪ If history reason is a Q.850 reason, it is translated to the SIP reason (according to the SIP-ISDN tables) and then to ISDN Redirect reason according to the table above.
UAS Behavior:
▪ History-Info is sent in the final response only.
▪ Upon receiving a request with History-Info, the UAS checks the policy in the request. If 'session', 'header', or 'history' policy tag is found, the (final) response is sent without History-Info. Otherwise, it is copied from the request.</td></tr>
<tr><td>Use Source Number as Display Name
<b>[UseSourceNumberAsDisplayName]</b></td><td>Applicable to Tel→IP calls.
No <b>[0]</b> = The Tel Source Number is used as the IP Source Number and the Tel Display Name is used as the IP Display Name (if Tel Display Name is received). If no Display Name is received from the Tel side, the IP Display Name remains empty (default).
Yes <b>[1]</b> = If a Tel Display Name is received, the Tel Source Number is used as the IP Source Number and the Tel Display Name is used as the IP Display Name. If no Display Name is received from the Tel side, the Tel Source Number is used as the IP Source Number and also as the IP Display Name.
Overwrite <b>[2]</b> = The Tel Source Number is used as the IP Source Number and also as the IP Display Name (even if the received Tel Display Name is not empty).</td></tr>
</table>

Table 5-3: Protocol Definition, General Parameters (continues on pages 64 to 70)

Parameter	Description
Use Display Name as Source Number **[UseDisplayNameAsSourceNumber]**	Applicable to IP→Tel calls. No **[0]** = The IP Source Number is used as the Tel Source Number and the IP Display Name is used as the Tel Display Name (if IP Display Name is received). If no Display Name is received from IP, the Tel Display Name remains empty (default). Yes **[1]** = If an IP Display Name is received, it is used as the Tel Source Number and also as the Tel Display Name, the Presentation is set to Allowed (0). If no Display Name is received from IP, the IP Source Number is used as the Tel Source Number and the Presentation is set to Restricted (1). For example: When the following is received 'from: 100 <sip:200@201.202.203.204>', the outgoing Source Number and Display Name are set to '100' and the Presentation is set to Allowed (0). When the following is received 'from: <sip:100@101.102.103.104>', the outgoing Source Number is set to '100' and the Presentation is set to Restricted (1).
Play Ringback Tone to IP **[PlayRBTone2IP]**	Don't Play **[0]** = Ringback tone isn't played to the IP side of the call (default). Play **[1]** = Ringback tone is played to the IP side of the call after SIP 183 session progress response is sent (for analog interfaces, this applies only to FXS modules; in FXO modules the Ringback tone isn't played). For digital modules: If configured to 1 ('Play'), and if EnableEarlyMedia = 1, for IP-to-Tel calls the gateway may play a ringback tone to IP, according to the following: ▪ For CAS interfaces, the gateway opens a voice channel, sends a 183+SDP response and plays a Ringback tone to IP. ▪ For ISDN interfaces, if a Progress or an Alert message with PI (1 or 8) is received from the ISDN, the gateway opens a voice channel, sends a 183+SDP or 180+SDP response, but it doesn't play a Ringback tone to IP. If PI (1 or 8) is received from the ISDN, the gateway assumes that Ringback tone is played by the ISDN Switch. Otherwise, the fateway plays a Ringback tone to IP after receiving an Alert message from the ISDN. It sends a 180+SDP response, signaling to the originating party to open a voice channel to hear the played Ringback tone. **Note 1:** To enable the gateway to send a 183/180+SDP responses, set EnableEarlyMedia to 1. **Note 2:** If EnableDigitDelivery = 1, the gateway doesn't play a Ringback tone to IP and doesn't send 183 or 180+SDP responses.
Play Ringback Tone to Tel **[PlayRBTone2Tel]**	Determines the method used to play Ringback tone to the Tel side. Don't Play **[0]** = Ringback Tone isn't played. Always Play **[1]** = Ringback Tone is played to the Tel side of the call when 180/183 response is received. Play According to 180/183 **[2]** = Ringback Tone is played to the Tel side of the call if no SDP is received in 180/183 responses. If 180/183 with SDP message is received, the gateway cuts through the voice channel and doesn't play Ringback tone (default). Play According to PI **[3]** = N/A.
Use Tgrp Information **[UseSIPTgrp]**	Disable **[0]** = Tgrp parameter isn't used (default). Send Only **[1]** = The trunk group number is added as the 'tgrp' parameter to the Contact header of outgoing SIP messages. If a trunk group number is not associated with the call, the 'tgrp' parameter isn't included. If a 'tgrp' value is specified in incoming messages, it is ignored. Send and Receive **[2]** = The functionality of outgoing SIP messages is identical to the functionality described in option (1). In addition, for incoming SIP messages, if the Request-URI includes a 'tgrp' parameter, the gateway routes the call according to that value (if possible). If the Contact header includes a 'tgrp' parameter, it is copied to the corresponding outgoing messages in that dialog.

Table 5-3: Protocol Definition, General Parameters (continues on pages 64 to 70)

Parameter	**Description**
Enable GRUU **[EnableGRUU]**	Determines whether or not the GRUU mechanism is used. Valid options include: Disable (default) = **[0]** Enable = **[1]**. The gateway obtains a GRUU by generating a normal REGISTER request. This request contains a Supported header field with the value "gruu". The gateway includes a "+sip.instance" Contact header field parameter for each contact for which the GRUU is desired. This Contact parameter contains a globally unique ID that identifies the gateway instance. The global unique id is as follows: ▪ If registration is per endpoint (AuthenticationMode=0), it is the MAC address of the gateway concatenated with the phone number of the endpoint. ▪ If the registration is per gateway (AuthenticationMode=1) it is only the MAC address. ▪ When the "User Information" mechanism is used, the globally unique ID is the MAC address concatenated with the phone number of the endpoint (defined in the User-Info file). If the Registrar/Proxy supports GRUU, the REGISTER responses contain the "gruu" parameter in each Contact header field. The Registrar/Proxy provides the same GRUU for the same AOR and instance-id in case of sending REGISTER again after expiration of the registration. The gateway places the GRUU in any header field which contains a URI. It uses the GRUU in the following messages: INVITE requests, 2xx responses to INVITE, SUBSCRIBE requests, 2xx responses to SUBSCRIBE, NOTIFY requests, REFER requests, and 2xx responses to REFER. **Note:** If the GRUU contains the "opaque" URI parameter, the gateway obtains the AOR for the user by stripping the parameter. The resulting URI is the AOR. For example: AOR: sip:alice@example.com GRUU: sip:alice@example.com;opaque="kjh29x97us97d"
User-Agent Information **[UserAgentDisplayInfo]**	Defines the string that is used in the SIP request header 'User-Agent' and SIP response header 'Server'. If not configured, the default string 'AudioCodes product-name s/w-version' is used (e.g., User-Agent: Audiocodes-Sip-Gateway-Mediant 1000 FXS/v.4.80.004.008). When configured, the string 'UserAgentDisplayInfo s/w-version' is used (e.g., User-Agent: MyNewOEM/v.4.80.004.008). Note that the version number can't be modified. The maximum string length is 50 characters.
Subject **[SIPSubject]**	Defines the value of the Subject header in outgoing INVITE messages. If not specified, the Subject header isn't included (default). The maximum length of the subject is limited to 50 characters.
Multiple Packetization Time Format **[MultiPtimeFormat]**	Determines whether the 'mptime' attribute is included in the outgoing SDP. Valid options include: 0 = Disable (default) 1 = Enable (includes the mptime attribute in the outgoing SDP -- PacketCable defined format) The 'mptime' attribute enables the gateway to define a separate Packetization period for each negotiated coder in the SDP. The 'mptime' attribute is only included if this parameter is enabled, even if the remote side includes it in the SDP offer. Upon reception, each coder receives its 'ptime' value in the following precedence: **1.** From 'mptime' attribute. **2.** From 'ptime' attribute. **3.** Default value.

Table 5-3: Protocol Definition, General Parameters (continues on pages 64 to 70)

Parameter	Description
Enable Reason Header **[EnableReasonHeader]**	Enables / disables the usage of the SIP Reason header. Disable **[0]**. Enable (default) **[1]**.
Enable Semi-Attended Transfer **[EnableSemiAttendedTransfer]**	Determines the gateway behavior when Transfer is initiated while still in Alerting state. Valid options include: Send REFER with Replaces (default) **[0]**. Send CANCEL, and after a 487 response is received, send REFER without Replaces **[1]**.
3xx Behavior **[3xxBehavior]**	Determines the gateway's behavior when a 3xx response is received for an outgoing INVITE request. The gateway can either use the same call identifiers (CallID, branch, to and from tags) or change them in the new initiated INVITE. 0 (forward) = Use different call identifiers for a redirected INVITE message (default). 1 (redirect) = Use the same call identifiers.
Retransmission Parameters	
SIP T1 Retransmission Timer [msec] **[SipT1Rtx]**	The time interval (in msec) between the first transmission of a SIP message and the first retransmission of the same message. The default is 500. **Note:** The time interval between subsequent retransmissions of the same SIP message starts with SipT1Rtx and is multiplied by two until SipT2Rtx. For example (assuming that SipT1Rtx = 500 and SipT2Rtx = 4000): The first retransmission is sent after 500 msec. The second retransmission is sent after 1000 (2*500) msec. The third retransmission is sent after 2000 (2*1000) msec. The fourth retransmission and subsequent retransmissions until SIPMaxRtx are sent after 4000 (2*2000) msec.
SIP T2 Retransmission Timer [msec] **[SipT2Rtx]**	The maximum interval (in msec) between retransmissions of SIP messages. The default is 4000. **Note:** The time interval between subsequent retransmissions of the same SIP message starts with SipT1Rtx and is multiplied by two until SipT2Rtx.
SIP Maximum Rtx **[SIPMaxRtx]**	Number of UDP transmissions (first transmission + retransmissions) of SIP messages. The range is 1 to 7. The default value is 7.

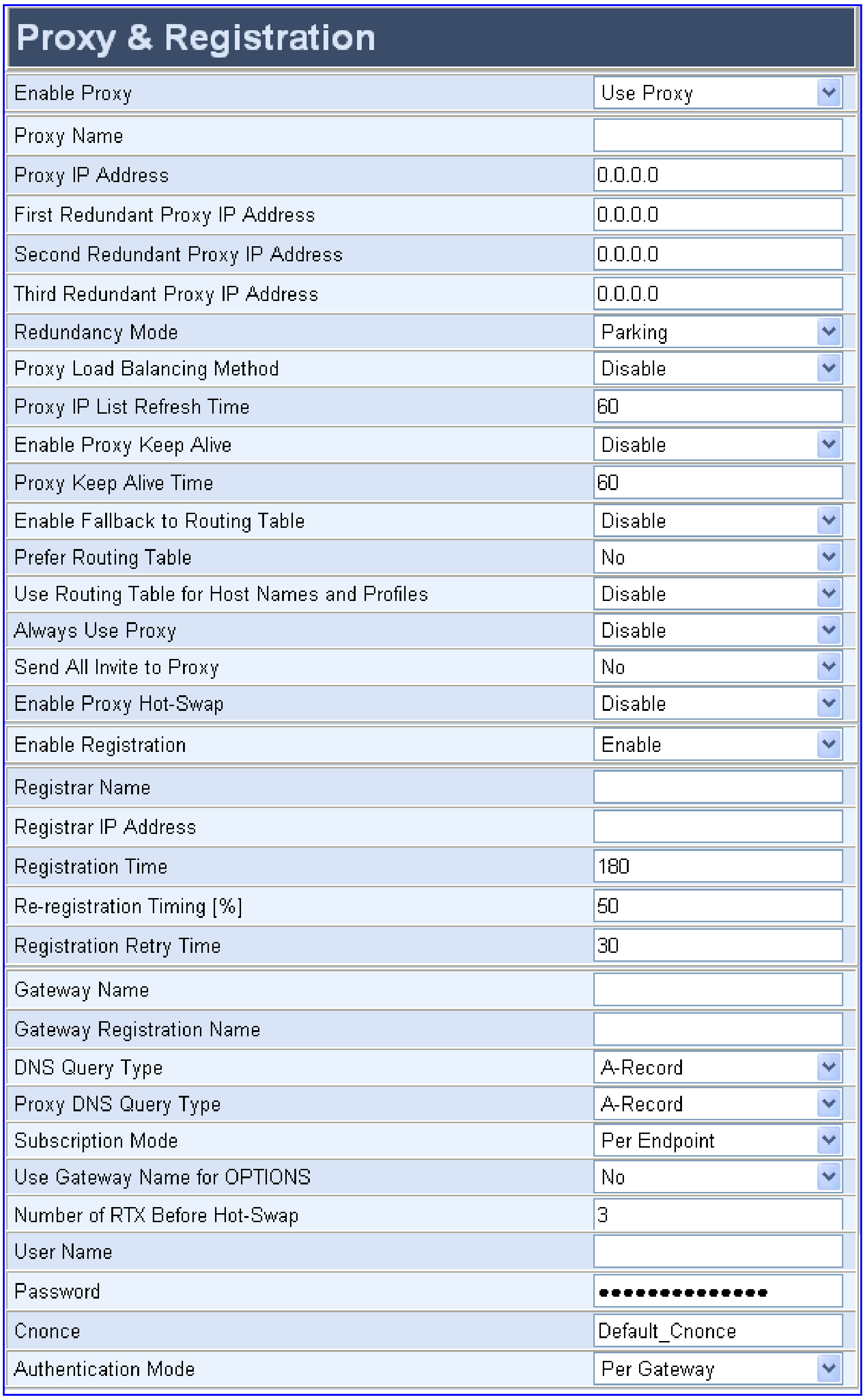
5.5.1.2 Proxy & Registration Parameters

The Proxy & Registration Parameters option is used to configure parameters that are associated with Proxy and Registration.

➢ To configure the Proxy & Registration parameters, take these 4 steps:

1. Open the 'Proxy & Registration' parameters screen (**Protocol Management** menu > **Protocol Definition** submenu > **Proxy & Registration** option); the 'Proxy & Registration' parameters screen is displayed.

Figure 5-6: Proxy & Registration Parameters Screen



Proxy & Registration

Enable Proxy	Use Proxy
Proxy Name	
Proxy IP Address	0.0.0.0
First Redundant Proxy IP Address	0.0.0.0
Second Redundant Proxy IP Address	0.0.0.0
Third Redundant Proxy IP Address	0.0.0.0
Redundancy Mode	Parking
Proxy Load Balancing Method	Disable
Proxy IP List Refresh Time	60
Enable Proxy Keep Alive	Disable
Proxy Keep Alive Time	60
Enable Fallback to Routing Table	Disable
Prefer Routing Table	No
Use Routing Table for Host Names and Profiles	Disable
Always Use Proxy	Disable
Send All Invite to Proxy	No
Enable Proxy Hot-Swap	Disable
Enable Registration	Enable
Registrar Name	
Registrar IP Address	
Registration Time	180
Re-registration Timing [%]	50
Registration Retry Time	30
Gateway Name	
Gateway Registration Name	
DNS Query Type	A-Record
Proxy DNS Query Type	A-Record
Subscription Mode	Per Endpoint
Use Gateway Name for OPTIONS	No
Number of RTX Before Hot-Swap	3
User Name	
Password	•••••••••••••••
Cnonce	Default_Cnonce
Authentication Mode	Per Gateway

2. Configure the Proxy & Registration parameters according to Table 5-4.
3. Click the **Submit** button to save your changes, or click the **Register** or **Un-Register** buttons to save your changes and to register / unregister to a Proxy / Registrar.
4. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Table 5-4: Proxy & Registration Parameters (continues on pages 72 to 78)

Parameter	Description
Enable Proxy **[IsProxyUsed]**	Don't Use Proxy **[0]** = Proxy isn't used, the internal routing table is used instead (default). Use Proxy **[1]** = Proxy is used. If you are using a Proxy server, enter the IP address of the primary Proxy server in the **Proxy IP address** field. If you are not using a Proxy server, you must configure the Tel to IP Routing table on the gateway (described in Section 5.5.4.2 on page 107).
Proxy Name **[ProxyName]**	Defines the Home Proxy Domain Name. If specified, the Proxy Name is used as Request-URI in REGISTER, INVITE and other SIP messages. If not specified, the Proxy IP address is used instead.
Proxy IP Address **[ProxyIP]**	IP address (and optionally port number) of the primary Proxy server you are using. Enter the IP address as FQDN or in dotted format notation (for example 201.10.8.1). You can also specify the selected port in the format: <IP Address>:<port>. This parameter is applicable only if you select 'Yes' in the 'Is Proxy Used' field. If you enable Proxy Redundancy (by setting EnableProxyKeepAlive=1 or 2), the gateway can work with up to four Proxy servers. If there is no response from the primary Proxy, the gateway tries to communicate with the redundant Proxies. When a redundant Proxy is found, the gateway either continues working with it until the next failure occurs or reverts to the primary Proxy (refer to the 'Redundancy Mode' parameter). If none of the Proxy servers respond, the gateway goes over the list again. The gateway also provides real time switching (hotswap mode), between the primary and redundant proxies ('IsProxyHotSwap=1'). If the first Proxy doesn't respond to INVITE message, the same INVITE message is immediately sent to the second Proxy. **Note 1:** If 'EnableProxyKeepAlive=1 or 2', the gateway monitors the connection with the Proxies by using keep-alive messages (OPTIONS or REGISTER). **Note 2:** To use Proxy Redundancy, you must specify one or more redundant Proxies using multiple 'ProxyIP= <IP address>' definitions. **Note 3:** When port number is specified (e.g., domain.com:5080), DNS NAPTR/SRV queries aren't performed, even if ProxyDNSQueryType is set to 1 or 2.
First Redundant Proxy IP Address **[ProxyIP]**	IP addresses of the first redundant Proxy you are using. Enter the IP address as FQDN or in dotted format notation (for example 192.10.1.255). You can also specify the selected port in the format: <IP Address>:<port>. **Note 1:** This parameter is available only if you select 'Use Proxy' in the 'Enable Proxy' field. **Note 2:** When port number is specified, DNS NAPTR/SRV queries aren't performed, even if ProxyDNSQueryType is set to 1 or 2. ***ini* file note:** The IP address of the first redundant Proxy is defined by the second repetition of the *ini* file parameter 'ProxyIP'.
Second Redundant Proxy IP Address **[ProxyIP]**	IP addresses of the second redundant Proxy you are using. Enter the IP address as FQDN or in dotted format notation (for example 192.10.1.255). You can also specify the selected port in the format: <IP Address>:<port>. **Note 1:** This parameter is available only if you select 'Use Proxy' in the 'Enable Proxy' field. **Note 2:** When port number is specified, DNS NAPTR/SRV queries aren't performed, even if ProxyDNSQueryType is set to 1 or 2. ***ini* file note:** The IP address of the second redundant Proxy is defined by the third repetition of the *ini* file parameter 'ProxyIP'.

Table 5-4: Proxy & Registration Parameters (continues on pages 72 to 78)

Parameter	**Description**
Third Redundant Proxy IP Address **[ProxyIP]**	IP addresses of the third redundant Proxy you are using. Enter the IP address as FQDN or in dotted format notation (for example 192.10.1.255). You can also specify the selected port in the format: <IP Address>:<port>. **Note 1:** This parameter is available only if you select 'Use Proxy' in the 'Enable Proxy' field. **Note 2:** When port number is specified, DNS NAPTR/SRV queries aren't performed, even if ProxyDNSQueryType is set to 1 or 2. ***ini* file note:** The IP addresses of the third redundant Proxy is defined by the fourth repetition of the *ini* file parameter 'ProxyIP'.
Redundancy Mode **[ProxyRedundancyMode]**	Parking **[0]** = Gateway continues working with the last active Proxy until the next failure (default). Homing **[1]** = Gateway always tries to work with the primary Proxy server (switches back to the main Proxy whenever it is available). **Note:** To use ProxyRedundancyMode, enable Keep-alive with Proxy option (EnableProxyKeepAlive=1 or 2).

Table 5-4: Proxy & Registration Parameters (continues on pages 72 to 78)

Parameter	Description
Proxy Load Balancing Method **[ProxyLoadBalancingMethod]**	Enables the usage of the Proxy Load Balancing mechanism. Valid options include: 0 = Load Balancing is disabled (default). 1 = Round Robin. 2 = Random Weights. When Round Robin (1) algorithm is used, a list of all possible Proxy IP addresses is compiled. This list includes all entries in the ProxyIP table after necessary DNS resolutions (including NAPTR and SRV, if configured). This list can handle up to 15 entries. After this list is compiled, the Proxy Keep-Alive mechanism (according to EnableProxyKeepAlive and ProxyKeepAliveTime) is used to mark each entry as Offline or Online. The balancing is only performed on Proxy servers that are marked as Online. All outgoing messages are equally distributed across the Proxy IP list. REGISTER messages are also distributed unless a RegistrarIP is configured. The Proxy IP list is refreshed according to ProxyIPListRefreshTime. If a change in the order of the entries in the list occurs, all load statistics are erased and balancing starts over again. When Random Weights (2) algorithm is used, the outgoing requests are not distributed equally among the Proxies. The weights are received from the DNS server by using SRV records. The gateway sends the requests in such a fashion that each Proxy receives a percentage of the requests according to its assigned weight. Load Balancing is not used in the following scenarios: ▪ The ProxyIP table includes more than one entry. ▪ The only Proxy defined is an IP address and not an FQDN. ▪ SRV usage is not enabled (DNSQueryType). ▪ The SRV response includes several records with a different Priority value.
Proxy IP List Refresh Time **[ProxyIPListRefreshTime]**	Defines the time interval (in seconds) between refreshes of the Proxy IP list. This parameter is used only when ProxyLoadBalancingMethod = 1. The interval range is 5 to 2,000,000. The default interval is 60.
Enable Proxy Keep Alive **[EnableProxyKeepAlive]**	Determines whether Keep-Alive with the Proxy is enabled or disabled. Valid options include: 0 = Disable (default) 1 = Enable Keep alive with Proxy using OPTIONS 2 = Enable Keep alive with Proxy using REGISTER If EnableProxyKeepAlive = 1, SIP OPTIONS message is sent every ProxyKeepAliveTime. If EnableProxyKeepAlive = 2, SIP REGISTER message is sent every RegistrationTime. Any response from the Proxy, either success (200 OK) or failure (4xx response) is considered as if the Proxy is correctly communicating. **Note 1:** This parameter must be set to 1 (OPTIONS) when Proxy redundancy is used. **Note 2:** When EnableProxyKeepAlive = 2 (REGISTER), the homing redundancy mode is disabled. **Note 3:** When the active proxy does not respond to INVITE messages sent by the gateway, the proxy is marked as offline. The behavior is similar to a Keep-Alive (OPTIONS or REGISTER) failure.
Proxy Keep Alive Time **[ProxyKeepAliveTime]**	Defines the Proxy keep-alive time interval (in seconds) between Keep-Alive messages. The default value is 60 seconds. **Note:** This parameter is applicable only if EnableProxyKeepAlive = 1 (OPTIONS). When EnableProxyKeepAlive = 2 (REGISTER), the time interval between Keep-Alive messages is determined by the parameter RegistrationTime.

Table 5-4: Proxy & Registration Parameters (continues on pages 72 to 78)

Parameter	**Description**
Enable Fallback to Routing Table **[IsFallbackUsed]**	No **[0]** = Gateway fallback is not used (default). Yes **[1]** = Internal Tel to IP Routing table is used when Proxy servers are not available. When the gateway falls back to the internal Tel to IP Routing table, the gateway continues scanning for a Proxy. When the gateway finds an active Proxy, it switches from internal routing back to Proxy routing. **Note:** To enable the redundant Proxies mechanism set 'EnableProxyKeepAlive' to 1 or 2.
Prefer Routing Table **[PreferRouteTable]**	Determines if the local Tel to IP routing table takes precedence over a Proxy for routing calls. No **[0]** = Only Proxy is used to route calls (default). Yes **[1]** = The gateway checks the 'Dest Phone Prefix' and/or 'Source Phone Prefix' field in the 'Tel to IP Routing' table for a match with the outgoing call. Only if a match is not found, a Proxy is used. **Note:** Applicable only if Proxy is not always used ('AlwaysSendToProxy' = 0, 'SendInviteToProxy' = 0).
Use Routing Table for Host Names and Profiles **[AlwaysUseRouteTable]**	Use the internal Tel to IP routing table to obtain the URI Host name and (optionally) an IP profile (per call), even if Proxy server is used. No **[0]** = Don't use (default). Yes **[1]** = Use. **Note:** This domain name is used, instead of Proxy name or Proxy IP address, in the INVITE SIP URI.
Always Use Proxy **[AlwaysSendToProxy]**	No **[0]** = Use standard SIP routing rules (default). Yes **[1]** = All SIP messages and Responses are sent to Proxy server. **Note:** Applicable only if Proxy server is used.
Send All INVITE to Proxy **[SendInviteToProxy]**	No **[0]** = INVITE messages, generated as a result of Transfer or Redirect, are sent directly to the URI (according to the Refer-To header in the REFER message or Contact header in 30x response) (default). Yes **[1]** = All INVITE messages, including those generated as a result of Transfer or Redirect are sent to Proxy. **Note:** Applicable only if Proxy server is used and 'AlwaysSendtoProxy=0'.
Enable Proxy Hot-Swap **[IsProxyHotSwap]**	Enable Proxy Hot-Swap redundancy mode. No **[0]** = Disabled (default). Yes **[1]** = Enabled. If Hot Swap is enabled, SIP INVITE message is first sent to the primary Proxy server. If there is no response from the primary Proxy server for 'Number of RTX before Hot-Swap' retransmissions, the INVITE message is resent to the redundant Proxy server.
Enable Registration **[IsRegisterNeeded]**	No **[0]** = Gateway doesn't register to Proxy / Registrar (default). Yes **[1]** = Gateway registers to Proxy / Registrar when the device is powered up and every RegistrationTime seconds. **Note:** The gateway sends a REGISTER request for each channel or for the entire gateway (according to the AuthenticationMode parameter).
Registrar Name **[RegistrarName]**	Registrar Domain Name. If specified, the name is used as Request-URI in REGISTER messages. If isn't specified (default), the Registrar IP address or Proxy name or Proxy IP address is used instead.
Registrar IP Address **[RegistrarIP]**	IP address and optionally port number of Registrar server. Enter the IP address in dotted format notation, for example, 201.10.8.1:<5080>. **Note 1:** If not specified, the REGISTER request is sent to the primary Proxy server (refer to 'Proxy IP address' parameter). **Note 2:** When port number is specified, DNS NAPTR/SRV queries aren't performed, even if DNSQueryType is set to 1 or 2.

Table 5-4: Proxy & Registration Parameters (continues on pages 72 to 78)

Parameter	Description
Registration Time **[RegistrationTime]**	Defines the time (in seconds) for which registration to a Proxy server is valid. The value is used in the header 'Expires'. In addition, this parameter defines the time interval between Keep-Alive messages when EnableProxyKeepAlive = 2 (REGISTER). Typically, a value of 3600 should be assigned for one hour registration. The gateway resumes registration according to the parameter RegistrationTimeDivider. The default value is 180. The valid range is 10 to 2000000.
Re-registration Timing (%) [**RegistrationTimeDivider**]	Defines the re-registration timing (in percentage). The timing is a percentage of the re-register timing set by the Registration server. The valid range is 50 to 100. The default value is 50. For example: If 'RegistrationTimeDivider = 70' (%) and Registration Expires time = 3600, the gateway resends its registration request after 3600 x 70% = 2520 sec.
Registration Retry Time **[RegistrationRetryTime]**	Defines the time period (in seconds) after which a Registration request is resent if registration fails with 4xx, or there is no response from the Proxy/Registrar. The default is 30 seconds. The range is 10 to 3600.
Gateway Name **[SIPGatewayName]**	Use this parameter to assign a name to the gateway (e.g., 'gateway1.com'). Ensure that the name you choose is the one that the Proxy is configured with to identify your media gateway. **Note:** If specified, the gateway Name is used as the host part of the SIP URI in the From header. If not specified, the gateway IP address is used instead (default).
Gateway Registration Name **[GWRegistrationName]**	Defines the user name that is used in From and To headers of REGISTER messages. Applicable only to single registration per gateway ('AuthenticationMode = 1). If 'GWRegistrationName' isn't specified (default), the 'Username' parameter is used instead. **Note:** This parameter is applicable only to a single gateway registration (AuthenticationMode = 1). When the gateway registers each channel separately (AuthenticationMode = 0), the user name is set to the channel's phone number.
DNS Query Type **[DNSQueryType]**	Enables the use of DNS Naming Authority Pointer (NAPTR) and Service Record (SRV) queries to resolve Proxy and Registrar servers and to resolve all domain names that appear in the Contact and Record-Route headers. Valid options include: ▪ 0 = A-Record (default) ▪ 1 = SRV ▪ 2 = NAPTR If set to A-Record, no NAPTR or SRV queries are performed. If set to SRV, and the Proxy / Registrar IP address parameter, the domain name in the Contact / Record-Route headers, or the IP address defined in the Routing tables contains a domain name without port definition, an SRV query is performed. The gateway uses the first host name received from the SRV query. The gateway then performs a DNS A-record query for the host name to locate an IP address. If set to NAPTR, an NAPTR query is performed. If it is successful, an SRV query is sent according to the information received in the NAPTR response. If the NAPTR query fails, an SRV query is performed according to the configured transport type. If the Proxy / Registrar IP address parameter, the domain name in the Contact / Record-Route headers, or the IP address defined in the Routing tables contains a domain name with port definition, the gateway performs a regular DNS A-record query. **Note:** To enable NAPTR/SRV queries for Proxy servers only, use the parameter ProxyDNSQueryType.

Table 5-4: Proxy & Registration Parameters (continues on pages 72 to 78)

Parameter	Description
Proxy DNS Query Type **[ProxyDNSQueryType]**	Enables the use of DNS Naming Authority Pointer (NAPTR) and Service Record (SRV) queries to discover Proxy servers. Valid options include: ▪ 0 = A-Record (default) ▪ 1 = SRV ▪ 2 = NAPTR If set to A-Record, no NAPTR or SRV queries are performed. If set to SRV and the Proxy IP address parameter contains a domain name without port definition (e.g., ProxyIP = domain.com), an SRV query is performed. The SRV query returns up to four Proxy host names and their weights. The gateway then performs DNS A-record queries for each Proxy host name (according to the received weights) to locate up to four Proxy IP addresses. Therefore, if the first SRV query returns two domain names, and the A-record queries return two IP addresses each, no more searches are performed. If set to NAPTR, an NAPTR query is performed. If it is successful, an SRV query is sent according to the information received in the NAPTR response. If the NAPTR query fails, an SRV query is performed according to the configured transport type. If the Proxy IP address parameter contains a domain name with port definition (e.g., ProxyIP = domain.com:5080), the gateway performs a regular DNS A-record query. **Note:** When enabled, NAPTR/SRV queries are used to discover Proxy servers even if the parameter DNSQueryType is disabled.
Enable SRV Queries **[EnableSRVQuery]**	This parameter is obsolete; use the parameter DNSQueryType.
Enable Proxy SRV Queries **[EnableProxySRVQuery]**	This parameter is obsolete; use the parameter ProxyDNSQueryType.
Subscription Mode **[SubscriptionMode]**	Determines the method the gateway uses to subscribe to an MWI server. Per Endpoint **[0]** = Each endpoint subscribes separately. This method is usually used for FXS modules (default). Per Gateway **[1]** = Single subscription for the entire gateway.
Use Gateway Name for OPTIONS **[UseGatewayNameForOptions]**	No **[0]** = Use the gateway's IP address in keep-alive OPTIONS messages (default). Yes **[1]** = Use 'GatewayName' in keep-alive OPTIONS messages. The OPTIONS Request-URI host part contains either the gateway's IP address or a string defined by the parameter 'Gatewayname'. The gateway uses the OPTIONS request as a keep-alive message to its primary and redundant Proxies (EnableProxyKeepAlive = 1).
Number of RTX Before Hot-Swap **[ProxyHotSwapRtx]**	Number of retransmitted INVITE messages before call is routed (hot swapped) to another Proxy. The valid range is 1 to 30. The default value is 3. **Note:** This parameter is also used for alternative routing using the Tel to IP Routing table. If a domain name in the routing table is resolved into 2 IP addresses, and if there is no response for 'ProxyHotSwapRtx' retransmissions to the INVITE message that is sent to the first IP address, the gateway immediately initiates a call to the second IP address.
User Name **[UserName]**	Username used for Registration and for Basic/Digest authentication process with Proxy / Registrar. Parameter doesn't have a default value (empty string). **Note 1:** Applicable only if single gateway registration is used ('Authentication Mode = Authentication Per gateway'). **Note 2:** The Authentication table can be used instead.
Password **[Password]**	Password used for Basic/Digest authentication process with Proxy / Registrar. Single password is used for all gateway ports. The default is 'Default_Passwd'. **Note:** The Authentication table can be used instead.

Table 5-4: Proxy & Registration Parameters (continues on pages 72 to 78)

Parameter	Description
Cnonce **[Cnonce]**	String used by the server and client to provide mutual authentication. (Free format i.e., 'Cnonce = 0a4f113b'). The default is 'Default_Cnonce'.
Authentication Mode **[AuthenticationMode]**	Per Endpoint **[0]** = Registration and Authentication separately for each endpoint. Per gateway **[1]** = Single Registration and Authentication for the gateway (default). Per Ch. Select Mode **[2]** = N/A. Authentication Per FXS Only **[3]** = Registration & Authentication only for FXS endpoints. Usually Authentication on a per endpoint basis is used for FXS modules, in which each endpoint registers (and authenticates) separately with its own username and password. Single Registration and Authentication (Authentication Mode=1) is usually defined for FXO and digital modules.

5.5.1.3 Coders

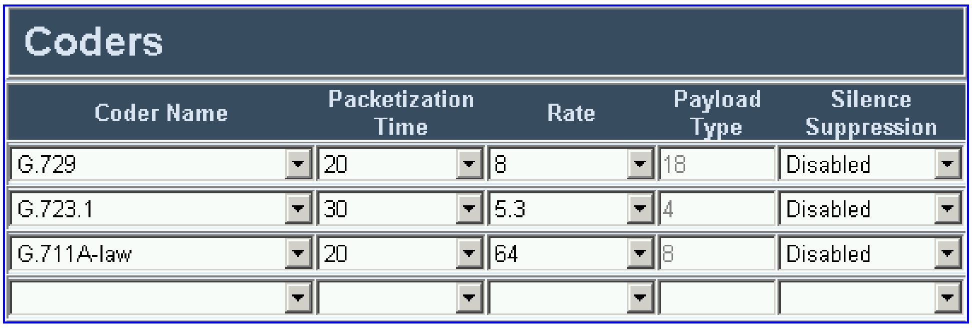
From the Coders screen you can configure the first to fifth preferred coders (and their attributes) for the gateway. The first coder is the highest priority coder and is used by the gateway whenever possible. If the far end gateway cannot use the coder assigned as the first coder, the gateway attempts to use the next coder and so forth.

You can also configure the Coders table using the *ini* file parameter CoderName (refer to Section 6.5.7 on page 230).

➢ To configure the gateway's coders, take these 9 steps:

1. Open the 'Coders' screen (**Protocol Management** menu > **Protocol Definition** submenu > **Coders** option); the 'Coders' screen is displayed.

Figure 5-7: Coders Screen



Coders

Coder Name	Packetization Time	Rate	Payload Type	Silence Suppression
G.729	20	8	18	Disabled
G.723.1	30	5.3	4	Disabled
G.711A-law	20	64	8	Disabled

2. From the Coder Name drop-down list, select the coder you want to use. For the full list of available coders and their corresponding attributes, refer to Table 5-5. **Note:** Each coder can appear only once.
3. From the Packetization Time drop-down list, select the packetization time (in msec) for the coder you selected. The packetization time determines how many coder payloads are combined into a single RTP packet.

   Note 1: If not specified, the ptime gets a default value.

   Note 2: The ptime specifies the packetization time the gateway expects to receive. The gateway always uses the ptime requested by the remote side for sending RTP packets.
4. From the Rate drop-down list, select the bit rate (in kbps) for the coder you selected.

5. In the 'Payload Type' field, if the payload type for the coder you selected is dynamic, enter a value from 0 to 120 (payload types of ‘well-known’ coders cannot be modified). The payload type identifies the format of the RTP payload. **Note:** If not specified, a default is used.

6. From the 'Silence Suppression' drop-down list, enable or disable the silence suppression option for the coder you selected.

   Note: For G.729 it is also possible to select silence suppression without adaptations.

7. Repeat steps 2 to 6 for the second to fifth coders (optional).

8. Click the **Submit** button to save your changes.

9. To permanently save the changes, refer to Section 5.9.2 on page 200.

Note 1: Only the ptime of the first coder in the defined coder list is declared in INVITE / 200 OK SDP, even if multiple coders are defined.

Note 2: If the coder G.729 is selected and silence suppression is disabled (for this coder), the gateway includes the string ‘annexb=no’ in the SDP of the relevant SIP messages. If silence suppression is enabled or set to ‘Enable w/o Adaptations’, ‘annexb=yes’ is included. An exception to this logic is when the remote gateway is a Cisco device (IsCiscoSCEMode).

Table 5-5: Supported Coders

Coder Name	Packetization Time	Rate	Payload Type	Silence Suppression
G.711 A-law [g711Alaw64k]	10, 20 (default), 30, 40, 50, 60, 80, 100, 120	Always 64	Always 8	Disable [0] Enable [1]
G.711 µ-law [g711Ulaw64k]	10, 20 (default), 30, 40, 50, 60, 80, 100, 120	Always 64	Always 0	Disable [0] Enable [1]
G.729 [g729]	10, 20 (default), 30, 40, 50, 60, 80, 100	Always 8	Always 18	Disable [0] Enable [1] Enable w/o Adaptations [2]
G.723.1 [g7231]	30 (default), 60, 90, 120	5.3 [0], 6.3 [1] (default)	Always 4	Disable [0] Enable [1]
G.726 [g726]	10, 20 (default), 30, 40, 50, 60, 80, 100, 120	16 [0], 24 [1], 32 [2] (default) 40 [3]	Dynamic (0-120)	Disable [0] Enable [1]
MS-GSM [gsmMS]	40 (default)	Always 13	Always 3	Disable [0] Enable [1]
NetCoder [NetCoder]	20 (default), 40, 60, 80, 100, 120	6.4 [0]; 7.2 [1] 8.0 [2] 8.8 [3] (default)	51 52 53 54	Disable [0] Enable [1]
Transparent [Transparent]	20 (default), 40, 60, 80, 100, 120	Always 64	Dynamic (0-120)	Disable [0] Enable [1]
T.38 [t38fax]	N/A	N/A	N/A	N/A
G.711A-law_VBD [g711AlawVbd]	10, 20 (default), 30, 40, 50, 60, 80, 100, 120	Always 64	Dynamic (0-120)	N/A
G.711U-law_VBD [g711UlawVbd]	10, 20 (default), 30, 40, 50, 60, 80, 100, 120	Always 64	Dynamic (0-120)	N/A

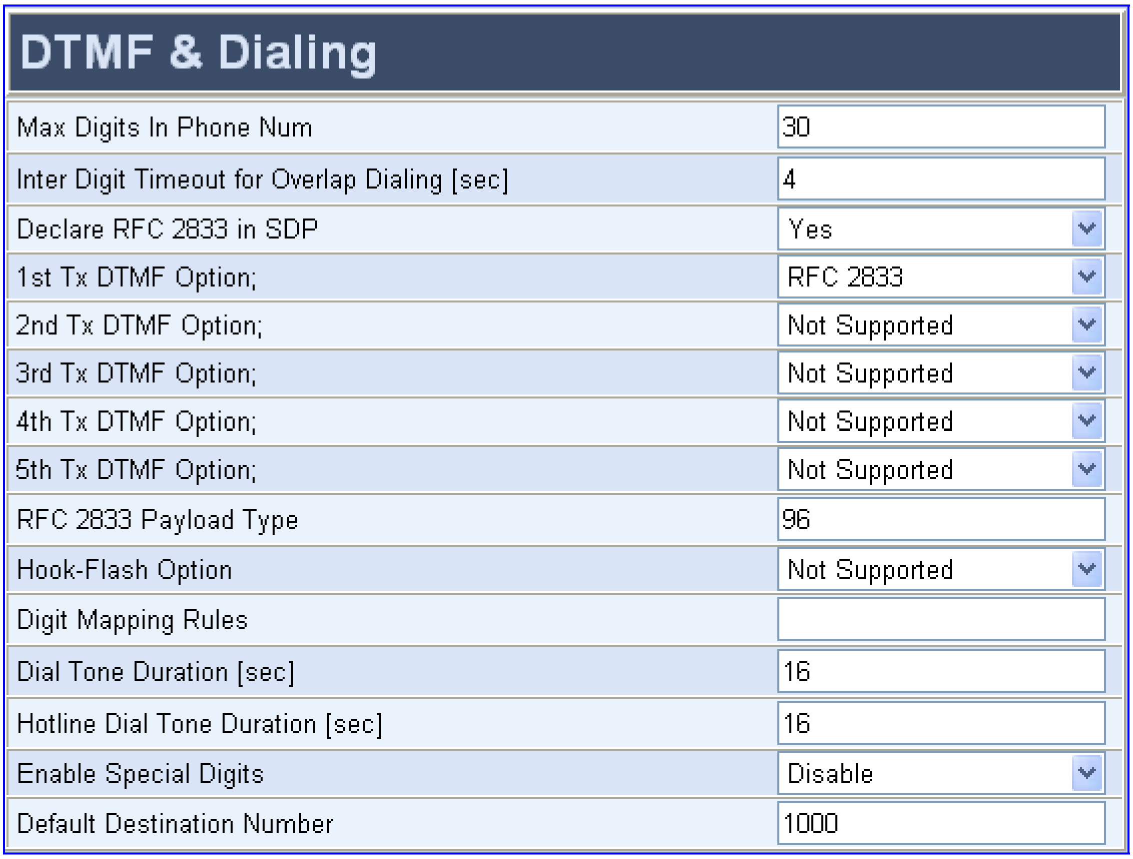
5.5.1.4 DTMF & Dialing Parameters

Use this screen to configure parameters that are associated with DTMF and dialing.

➢ **To configure the dialing parameters, take these 4 steps:**

1. Open the 'DTMF & Dialing' screen (**Protocol Management** menu > **Protocol Definition** submenu > **DTMF & Dialing** option); the 'DTMF & Dialing' screen is displayed.

Figure 5-8: DTMF & Dialing Screen



DTMF & Dialing

Max Digits In Phone Num	30
Inter Digit Timeout for Overlap Dialing [sec]	4
Declare RFC 2833 in SDP	Yes
1st Tx DTMF Option;	RFC 2833
2nd Tx DTMF Option;	Not Supported
3rd Tx DTMF Option;	Not Supported
4th Tx DTMF Option;	Not Supported
5th Tx DTMF Option;	Not Supported
RFC 2833 Payload Type	96
Hook-Flash Option	Not Supported
Digit Mapping Rules	
Dial Tone Duration [sec]	16
Hotline Dial Tone Duration [sec]	16
Enable Special Digits	Disable
Default Destination Number	1000

2. Configure the DTMF & Dialing parameters according to Table 5-6.
3. Click the **Submit** button to save your changes.
4. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Table 5-6: DTMF & Dialing Parameters (continues on pages 80 to 82)

Parameter	Description
Max Digits in Phone Num **[MaxDigits]**	Defines the maximum number of collected destination number digits received from the Tel side (ISDN uses overlap dialing). When the number of collected digits reaches the maximum, the gateway uses these digits for the called destination number. The valid range is 1 to 49. The default value is 30. **Note 1:** Dialing ends when the maximum number of digits is dialed, the Interdigit Timeout expires, the '#' key is dialed, or a digit map pattern is matched. **Note 2:** Digit Mapping Rules can be used instead.

Table 5-6: DTMF & Dialing Parameters (continues on pages 80 to 82)

Parameter	**Description**
Inter Digit Timeout for Overap Dialing [sec] **[TimeBetweenDigits]**	Defines the time (in seconds) that the gateway waits between digits that are received from the Tel side (ISDN uses overlap dialing). When this inter-digit timeout expires, the gateway uses the collected digits for the called destination number. The valid range is 1 to 10 seconds. The default value is 4 seconds.
Declare RFC 2833 in SDP **[RxDTMFOption]**	Defines the supported Receive DTMF negotiation method. No **[0]** = Don't declare RFC 2833 telephony-event parameter in SDP Yes **[3]** = Declare RFC 2833 telephony-event parameter in SDP (default) The Mediant 1000 is designed to always be receptive to RFC 2833 DTMF relay packets. Therefore, it is always correct to include the 'telephony-event' parameter as a default in the SDP. However some gateways use the absence of the 'telephony-event' from the SDP to decide to send DTMF digits in-band using G.711 coder, if this is the case you can set 'RxDTMFOption=0'.
1st to 5th Tx DTMF Option **[TxDTMFOption]**	Determines a single or several preferred transmit DTMF negotiation methods. 0 (Not Supported) = No negotiation, DTMF digits are sent according to the parameters 'DTMFTransportType' and 'RFC2833PayloadType' (default). 1 (INFO Nortel) = Sends DTMF digits according to IETF <draft-choudhuri-sip-info-digit-00>. 2 (NOTIFY) = Sends DTMF digits according to <draft-mahy-sipping-signaled-digits-01>. 3 (INFO Cisco) = Sends DTMF digits according to Cisco format. 4 (RFC 2833). **Note 1:** DTMF negotiation methods are prioritized according to the order of their appearance. **Note 2:** When out-of-band DTMF transfer is used (1, 2 or 3), the parameter 'DTMFTransportType' is automatically set to 0 (DTMF digits are erased from the RTP stream). **Note 3:** When RFC 2833 (4) is selected, the gateway: ▪ Negotiates RFC 2833 Payload Type (PT) using local and remote SDPs. ▪ Sends DTMF packets using RFC 2833 PT according to the PT in the received SDP. ▪ Expects to receive RFC 2833 packets with the same PT as configured by the parameter 'RFC2833PayloadType'. ▪ Uses the same PT for send and receive if the remote party doesn't include the RFC 2833 DTMF PT in its SDP. **Note 4:** When TxDTMFOption is set to 0, the RFC 2833 PT is set according to the parameter 'RFC2833PayloadType' for both transmit and receive. ***ini* file note:** The DTMF transmit methods are defined using a repetition of the same (TxDTMFOption) parameter (up to five options can be provided).
RFC 2833 Payload Type **[RFC2833PayloadType]**	The RFC 2833 DTMF relay dynamic payload type. Range: 96 to 99, 106 to 127; Default = 96 The 100, 102 to 105 range is allocated for proprietary usage. **Note 1:** Cisco is using payload type 101 for RFC 2833. **Note 2:** When RFC 2833 payload type (PT) negotiation is used (TxDTMFOption=4), this payload type is used for the received DTMF packets. If negotiation isn't used, this payload type is used for receive and for transmit.
Hook Flash Option **[HookFlashOption]**	Supported hook-flash Transport Type (method by which hook-flash is sent and received). Valid options include: 0 = Hook-Flash indication isn't sent (default) 1 = Send proprietary INFO message with Hook-Flash indication 4 **=** RFC 2833
Use INFO for Hook-Flash **[IsHookFlashUsed]**	This parameter is obsolete; use instead the parameter HookFlashOption.

Table 5-6: DTMF & Dialing Parameters (continues on pages 80 to 82)

Parameter	Description
Digit Mapping Rules **[DigitMapping]**	Digit map pattern used to reduce the dialing period when Overlap dialing is used. If the digit string (dialed number) has matched one of the patterns in the digit map, the gateway stops collecting digits and starts to establish a call with the collected number The digit map pattern contains up to 52 options separated by a vertical bar (|) and enclosed in parenthesis. The maximum length of the entire digit pattern is limited to 152 characters. Available notations: ▪ [n-m] represents a range of numbers ▪ '.' (single dot) represents repetition ▪ 'x' represents any single digit ▪ 'T' represents a dial timer (configured by TimeBetweenDigits parameter) ▪ 'S' should be used when a specific rule, that is part of a general rule, is to be applied immediately. For example, if you enter the general rule x.T and the specific rule 11x, you should append 'S' to the specific rule 11xS. For example: (11xS|00T|[1-7]xxx|8xxxxxxx|#xxxxxxx|*xx|91xxxxxxxxxx|9011x.T) **Note:** The digitmap mechanism is applicable only when ISDN Overlap dialing is used (ISDNRxOverlap = 1).
Dial Tone Duration [sec] **[TimeForDialTone]**	▪ **Analog:** Duration (in seconds) that the dial tone is played. The default time is 16 seconds. FXS module ports play the dial tone after the phone is picked up (off hook); while FXO module ports play the dial tone after port is seized in response to ringing. **Note 1:** During play of dial tone, the gateway waits for DTMF digits. **Note 2:** 'TimeForDialTone' is not applicable when Automatic Dialing is enabled. ▪ **Digital:** Duration (in seconds) of the dial tone played to an ISDN terminal. Applicable to overlap dialing when 'ISDNInCallsBehavior = 65536'. The dial tone is played if the ISDN Setup message doesn't include the called number. The valid range is 0 to 60. The default time is 16 seconds.
Hotline Dial Tone Duration **[HotLineToneDuration]**	Duration (in seconds) of the Hotline dial tone. If no digits are received during the Hotline dial tone duration, the gateway initiates a call to a preconfigured number (set in the automatic dialing table). The valid range is 0 to 60. The default time is 16 seconds. **Note:** Applicable to FXS and FXO modules.
Enable Special Digits **[IsSpecialDigits]**	Disable **[0]** = '*' or '#' terminate number collection (default). Enable **[1]** = if you want to allow '*' and '#' to be used for telephone numbers dialed by a user or entered for the endpoint telephone number. **Note:** The # and * can always be used as first digit of a dialed number, even if you select 'Disable' for this parameter.
Default Destination Number **[DefaultNumber]**	Defines the telephone number that the gateway uses if the parameter 'TrunkGroup_x' doesn't include a phone number. The parameter is used as a starting number for the list of channels comprising all trunk groups in the gateway.

5.5.2 Configuring the Advanced Parameters

Use this submenu to configure the following gateway’s advanced control protocol parameters:

- General Parameters (refer to Section 5.5.2.1 on page 83)
- Supplementary Services (refer to Section 5.5.2.2 on page 91)
- Metering Tones (refer to Section 5.5.2.3 on page 95)
- Keypad Features (refer to Section 5.5.2.4 on page 97)

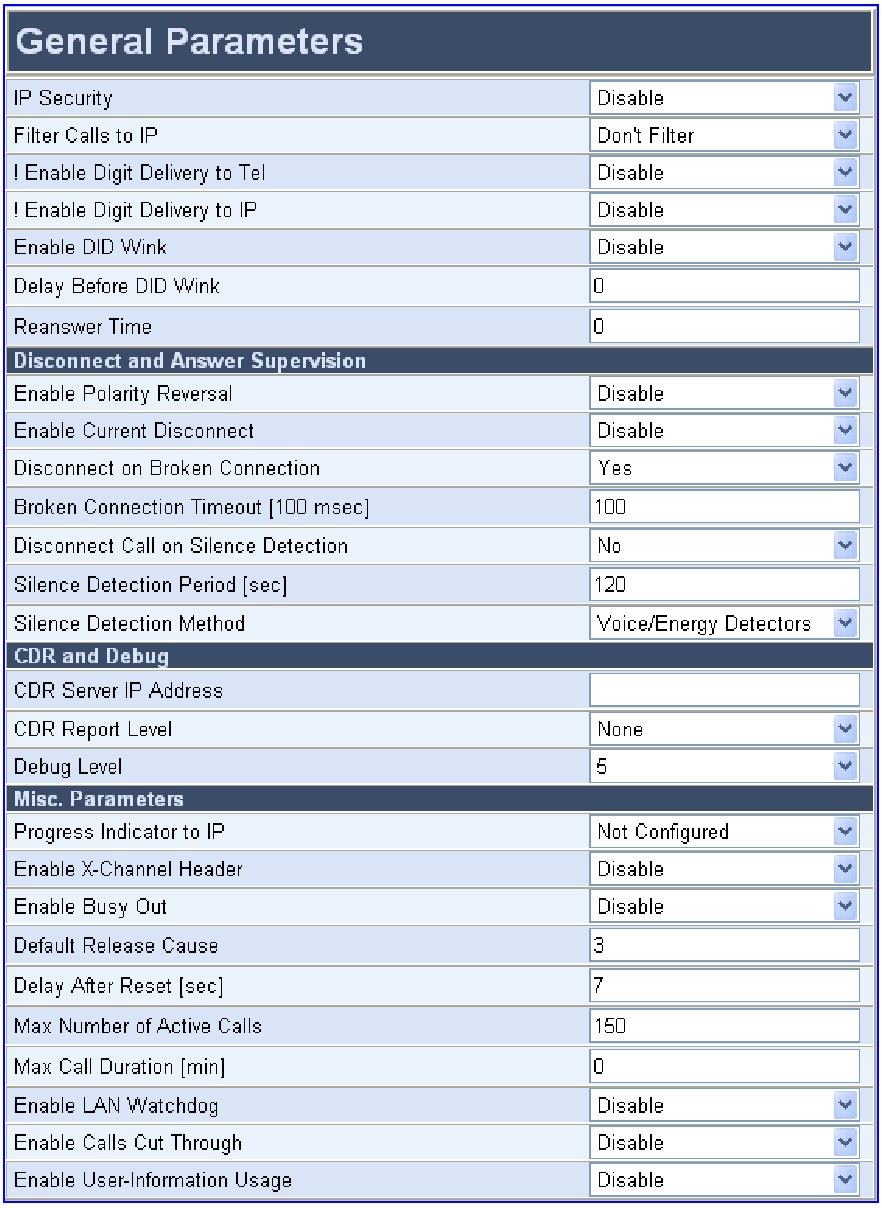
5.5.2.1 General Parameters

Use this screen to configure general control protocol parameters.

➢ **To configure the general parameters under Advanced Parameters, take these 4 steps:**

1. Open the ‘General Parameters’ screen (**Protocol Management** menu > **Advanced Parameters** submenu > **General Parameters** option); the ‘General Parameters’ screen is displayed.

Figure 5-9: Advanced Parameters, General Parameters Screen



General Parameters

IP Security	Disable
Filter Calls to IP	Don't Filter
! Enable Digit Delivery to Tel	Disable
! Enable Digit Delivery to IP	Disable
Enable DID Wink	Disable
Delay Before DID Wink	0
Reanswer Time	0
Disconnect and Answer Supervision	
Enable Polarity Reversal	Disable
Enable Current Disconnect	Disable
Disconnect on Broken Connection	Yes
Broken Connection Timeout [100 msec]	100
Disconnect Call on Silence Detection	No
Silence Detection Period [sec]	120
Silence Detection Method	Voice/Energy Detectors
CDR and Debug	
CDR Server IP Address	
CDR Report Level	None
Debug Level	5
Misc. Parameters	
Progress Indicator to IP	Not Configured
Enable X-Channel Header	Disable
Enable Busy Out	Disable
Default Release Cause	3
Delay After Reset [sec]	7
Max Number of Active Calls	150
Max Call Duration [min]	0
Enable LAN Watchdog	Disable
Enable Calls Cut Through	Disable
Enable User-Information Usage	Disable

2. Configure the general parameters under 'Advanced Parameters' according to Table 5-7.
3. Click the **Submit** button to save your changes.
4. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Table 5-7: Advanced Parameters, General Parameters (continues on pages 85 to 90)

Parameter	Description
IP Security **[SecureCallsFromIP]**	No **[0]** = Gateway accepts all SIP calls (default). Yes **[1]** = Gateway accepts SIP calls only from IP addresses defined in the Tel to IP routing table. The gateway rejects all calls from unknown IP addresses. For detailed information on the Tel to IP Routing table, refer to Section 5.5.4.2 on page 107. **Note:** Specifying the IP address of a Proxy server in the Tel to IP Routing table enables the gateway to only accept calls originating in the Proxy server and rejects all other calls.
Filter Calls to IP **[FilterCalls2IP]**	Don't Filter **[0]** = Disabled (default) Filter **[1]** = Enabled If the filter calls to IP feature is enabled, then when a Proxy is used, the gateway first checks the Tel→IP routing table before making a call through the Proxy. If the number is not allowed (number isn't listed or a Call Restriction routing rule, IP = 0.0.0.0 is applied), the call is released.
Enable Digit Delivery to IP **[EnableDigitDelivery2IP]**	Disable **[0]** = Disabled (default). Enable **[1]** = Enable digit delivery to IP. The digit delivery feature enables sending of DTMF digits to the destination IP address after the Tel→IP call was answered. To enable this feature, modify the called number to include at least one 'p' character. The gateway uses the digits before the 'p' character in the initial INVITE message. After the call was answered the gateway waits for the required time (# of 'p' * 1.5 seconds) and then sends the rest of the DTMF digits using the method chosen (in-band, out-of-band). **Note:** The called number can include several 'p' characters (1.5 seconds pause). For example, the called number can be as follows: 1001pp699, 8888p9p300.

Table 5-7: Advanced Parameters, General Parameters (continues on pages 85 to 90)

Parameter	Description
Enable Digit Delivery to Tel **[EnableDigitDelivery]**	Disable **[0]** = Disabled (default). Enable **[1]** = Enable Digit Delivery feature for Mediant 1000. The digit delivery feature enables sending of DTMF digits to the gateway's port after the call was answered. For IPàTel calls, after answering the call, the Mediant 1000 plays the DTMF digits (of the called number) towards the phone line. **Note 1:** The called number can include characters 'p' (1.5 seconds pause) and 'd' (detection of dial tone). If character 'd' is used, it must be the first 'digit' in the called number. The character 'p' can be used several times. For example (for FXS/FXO module), the called number can be as follows: d1005, dpp699, p9p300. To add the 'd' and 'p' digits, use the usual number manipulation rules. **Note 2:** To use this feature with FXO modules, configure the gateway to operate in one stage dialing mode. **Note 3:** If the parameter 'EnableDigitDelivery' is enabled, it is possible to configure the FXS/FXO gateway to wait for dial tone per destination phone number (before or during dialing of destination phone number), therefore, the parameter 'IsWaitForDialTone' (configurable for the entire gateway) is ignored. **Note 4:** For digital modules, if the called number in IP-to-Tel call includes the characters 'w' or 'p', the gateway places a call with the first part of the called number (before 'w' or 'p') , and plays DTMF digits after the call is answered. If the character 'w' is used, the gateway waits for detection of dial tone before it starts playing DTMF digits. For example, if the number '1007766p100' is defined as the called number, the gateway places a call with 1007766 as the destination number, then, after the call is answered, it waits for 1.5 seconds and plays the rest of the number (100) as DTMF digits. Other examples: 1664wpp102, 66644ppp503, 7774w100pp200. **Note 5:** The FXS and digital modules send SIP 200 OK responce only after the DTMF dialing has completed.
Enable DID Wink **[EnableDIDWink]**	Disable **[0]** = DID is disabled (default). Enable **[1]** = Enable DID. If enabled, the Mediant 1000 can be used for connection to EIA/TIA-464B DID Loop Start lines. Both FXO (detection) and FXS (generation) are supported. An FXO module dials DTMF digits after a Wink signal is detected (instead of a Dial tone). An FXS module generates the Wink signal after the detection of offhook (instead of playing a Dial tone).
Delay Before DID Wink **[DelayBeforeDIDWink]**	Defines the time interval (in seconds) between detection of offhook and generation of DID Wink. Applicable only to FXS modules. The valid range is 0 to 1,000. The default value is 0.
Reanswer Time **[RegretTime]**	▪ **Analog:** The time period after user hangs up the phone and before call is disconnected (FXS). Also called regret time. ▪ **Digital:** Determines the time period (in seconds) the gateway waits for an MFC R2 Resume (Reanswer) signal once a Suspend (Clear back) signal was received from the PBX. If this timer expires, the call is released.. Applicable only for MFC R2 CAS Brazil variant. The valid range is 0 to 255 (in seconds). The default value is 0.

Table 5-7: Advanced Parameters, General Parameters (continues on pages 85 to 90)

Parameter	Description
Disconnect and Answer Supervision	
Enable Polarity Reversal **[EnableReversalPolarity]**	Disable **[0]** = Disable the polarity reversal service (default). Enable **[1]** = Enable the polarity reversal service. If the polarity reversal service is enabled, then the FXS module changes the line polarity on call answer and changes it back on call release. The FXO module sends a 200 OK response when polarity reversal signal is detected (applicable to one stage dialing only), and releases a call when a second polarity reversal signal is detected.
Enable Current Disconnect **[EnableCurrentDisconnect]**	Disable **[0]** = Disable the current disconnect service (default). Enable **[1]** = Enable the current disconnect service. If the current disconnect service is enabled, the FXO module releases a call when current disconnect signal is detected on its port, while the FXS module generates a 'Current Disconnect Pulse' after a call is released from IP. The current disconnect duration is determined by the parameter 'CurrentDisconnectDuration'. The current disconnect threshold (FXO only) is determined by the parameter 'CurrentDisconnectDefaultThreshold'. The frequency at which the analog line voltage is sampled is determined by the parameter 'TimeToSampleAnalogLineVoltage'.
Disconnect on Broken Connection **[DisconnectOnBrokenConnection]**	No **[0]** = Don't release the call. Yes **[1]** = Call is released if RTP packets are not received for a predefined timeout (default). **Note 1:** If enabled, the timeout is set by the parameter 'BrokenConnectionEventTimeout', in 100 msec resolution. The default timeout is 10 seconds: (BrokenConnectionEventTimeout =100). **Note 2:** This feature is applicable only if RTP session is used without Silence Compression. If Silence Compression is enabled, the gateway doesn't detect that the RTP connection is broken. **Note 3:** During a call, if the source IP address (from where the RTP packets were sent) is changed without notifying the gateway, the gateway filters these RTP packets. To overcome this issue, set 'DisconnectOnBrokenConnection=0'; the gateway doesn't detect RTP packets arriving from the original source IP address, and switches (after 300 msec) to the RTP packets arriving from the new source IP address.
Broken Connection Timeout **[BrokenConnectionEventTimeout]**	The amount of time (in 100 msec units) an RTP packet isn't received, after which a call is disconnected. The valid range is 1 to 1000. The default value is 100 (10 seconds). **Note 1:** Applicable only if 'DisconnectOnBrokenConnection = 1'. **Note 2:** Currently this feature works only if Silence Suppression is disabled.
Disconnect Call on Silence Detection **[EnableSilenceDisconnect]**	Yes **[1]** = The gateway disconnect calls in which silence occurs in both (call) directions for more than 120 seconds. No **[0]** = Call is not disconnected when silence is detected (default). The silence duration can be set by the 'FarEndDisconnectSilencePeriod' parameter (default 120). **Note:** To activate this feature set: 'EnableSilenceCompression' to 1 and 'FarEndDisconnectSilenceMethod' to 1.
Silence Detection Period [sec] **[FarEndDisconnectSilencePeriod]**	Duration of silence period (in seconds) prior to call disconnection. The range is 10 to 28,800 (i.e., 8 hours). The default is 120 seconds.
Silence Detection Method **[FarEndDisconnectSilenceMethod]**	Silence detection method. None **[0] =** Silence detection option is disabled. Packets Count **[1]** = According to packet count. Voice/Energy Detectors **[2]** = N/A. All **[3]** = N/A.

Table 5-7: Advanced Parameters, General Parameters (continues on pages 85 to 90)

Parameter	Description
Send Digit Pattern on Connect **[TelConnectCode]**	Defines a digit pattern that is sent to the Tel side after 200 OK is received from the IP side. The digit pattern is a predefined DTMF sequence that is used to indicate an answer signal (e.g., for billing purposes). Applicable only to FXS modules. The valid range is 1 to 8 characters.
CDR and Debug	
CDR Server IP Address **[CDRSyslogServerIP]**	Defines the destination IP address for CDR logs. The default value is a null string that causes the CDR messages to be sent with all Syslog messages. **Note:** The CDR messages are sent to UDP port 514 (default Syslog port).
CDR Report Level **[CDRReportLevel]**	Determines whether or not CDRs are sent to the Syslog server, and if enabled, at which events they are sent. Valid options include: 0 = Call Detail Record (CDR) is not used. 1 = CDR is sent to the Syslog server at the end of each call. 2 = CDR report is sent to Syslog at the start and end of each call. 3 = CDR report is sent to Syslog at connection and at the end of each call. The CDR Syslog message complies with RFC 3161 and is identified by: Facility = 17 (local1) and Severity = 6 (Informational).
Debug Level **[GwDebugLevel]**	Syslog logging level. One of the following debug levels can be selected: 0 **[0]** = Debug is disabled (default) 1 **[1]** = Flow debugging is enabled 2 **[2]** = Flow and device interface debugging are enabled 3 **[3]** = Flow, device interface and stack interface debugging are enabled 4 **[4]** = Flow, device interface, stack interface and session manager debugging are enabled 5 **[5]** = Flow, device interface, stack interface, session manager and device interface expanded debugging are enabled. **Note:** Usually set to 5 if debug traces are needed.
Misc. Parameters	
Progress Indicator to IP **[ProgressIndicator2IP]**	▪ **Analog (FXS/FXO) modules:** No PI **[0]** = For Tel→IP calls, the gateway sends ‘180 Ringing’ SIP response to IP after placing a call to phone (FXS) or to PBX (FXO). PI = 1, PI = 8 **[1]**, **[8]** = For Tel→IP calls, if ‘EnableEarlyMedia=1’, the gateway sends ‘183 session in progress’ message + SDP, immediately after a call is placed to Phone/PBX. This is used to cut through the voice path, before remote party answers the call, enabling the originating party to listen to network Call Progress Tones (such as Ringback tone or other network announcements). Not Configured **[-1]** = Default values are used. The default for FXO modules is 1; The default for FXS modules is 0. ▪ **Digital (ISDN/CAS) modules:** Not configured **[-1]** = for ISDN spans, the progress indicator (PI) that is received in ISDN Proceeding, Progress and Alert messages is used as described in the options below (default). No P **[0]** = For IP→Tel call, the gateway sends ‘180 Ringing’ SIP response to IP after receiving ISDN Alert or (for CAS) after placing a call to PBX/PSTN. PI = 1, PI = 8; For IP→Tel call, if ‘EnableEarlyMedia=1’, the gateway sends ‘180 Ringing’ with SDP in response to an ISDN alert, or it sends a ‘183 session in progress’ message with SDP in response to only the first received ISDN Proceeding or Progress message, after a call is placed to PBX/PSTN over the trunk.

Table 5-7: Advanced Parameters, General Parameters (continues on pages 85 to 90)

Parameter	Description
Enable X-Channel Header **[XChannelHeader]**	0 = x-channel header is not used (default). 1 = x-channel header is generated, with trunk/B-channel information. The header provides information on the E1/T1 physical trunk/B-channel on which the call is received or placed. For example 'x-channel: DS/DS1-5/22', where 'DS/DS-1' is a constant string, '5' is the trunk number, and '22' is the B-channel. This header is generated by the gateway and is sent in the following messages: INVITE and 183/180/200OK responses.
Enable Busy Out **[EnableBusyOut]**	No **[0]** = 'Busy out' feature is not used (default). Yes **[1]** = 'Busy out' feature is enabled. When Busy out is enabled and certain scenarios exist, the gateway performs a specific behavior depending whether it's an analog or digital gateway: ▪ Analog gateways: a reorder tone (determined by FXSOOSBehavior) is played when the phone is offhooked. ▪ Digital gateways: all E1/T1 trunks are automatically put out of service by sending a remote alarm (AIS) or Service Out message for T1 PRI trunks that support these messages (NI-2, 4/5-ESS, DMS-100 and Meridian) These behaviors are performed due to one of the following scenarios: ▪ Physically disconnected from the network (i.e., Ethernet cable is disconnected). ▪ The Ethernet cable is connected, but the gateway can't communicate with any host. Note that LAN Watch-Dog must be activated (EnableLANWatchDog = 1). ▪ The gateway can't communicate with the gatekeeper/proxy (according to the Proxy keep-alive mechanism) and no other alternative exists to send the call. **Notes for Analog gateways:** ▪ The FXSOOSBehavior parameter is used to control the behavior of the FXS endpoints of the gateway when a Busy out or Graceful Lock occurs. ▪ FXO endpoints during Busy out and Lock are inactive. ▪ Refer to LifeLineType parameter for complementary optional behavior. **Note for Digtal gateways:** The Busy out behavior varies between different protocol types.
Default Release Cause **[DefaultReleaseCause]**	Default Release Cause (to IP) for IP→Tel calls, used when the gateway initiates a call release, and if an explicit matching cause for this release isn't found, a default release cause can be configured: The default release cause is: NO_ROUTE_TO_DESTINATION (3). Other common values are: NO_CIRCUIT_AVAILABLE (34), DESTINATION_OUT_OF_ORDER (27), etc. **Note:** The default release cause is described in the Q.931 notation, and is translated to corresponding SIP 40x or 50x value. For example: 404 for 3, 503 for 34 and 502 for 27 For mapping of SIP to Q.931 and Q.931 to SIP release causes, refer to H on page 445.
Delay After Reset [sec] **[GWAppDelayTime]**	Defines the amount of time (in seconds) the gateway's operation is delayed after a reset cycle. The valid range is 0 to 45. The default value is 7 seconds. **Note:** This feature helps to overcome connection problems caused by some LAN routers or IP configuration parameters change by a DHCP Server.
Max Number of Active Calls **[MaxActiveCalls]**	Defines the maximum number of calls that the gateway can have active at the same time. If the maximum number of calls is reached, new calls are not established. The default value is max available channels (no restriction on the maximum number of calls). The valid range is 1 to max number of channels.

Table 5-7: Advanced Parameters, General Parameters (continues on pages 85 to 90)

Parameter	Description
Max Call Duration (sec) **[MaxCallDuration]**	Defines the maximum call duration in seconds. If this time expires, both sides of the call are released (IP and Tel). The valid range is 0 to 120. The default is 0 (no limitation).
Enable LAN Watchdog **[EnableLanWatchDog]**	Disable **[0]** = Disable LAN Watch-Dog (default). Enable **[1]** = Enable LAN Watch-Dog. When LAN Watch-Dog is enabled, the gateway's overall communication integrity is checked periodically. If no communication for about 3 minutes is detected, the gateway performs a self test. ▪ If the self test succeeds, the problem is logical link down (i.e. Ethernet cable disconnected on the switch side), and the Busy out mechanism is activated if enabled (EnableBusyOut = 1). ▪ If the self test fails, the gateway restarts to overcome internal fatal communication error. **Note:** Enable LAN Watchdog is relevant only if the Ethernet connection is full duplex.
Enable Calls Cut Through **[CutThrough]**	Enables users to receive incoming IP calls while the port is in an offhooked state. Disable **[0]** = Disabled (default). Enable **[1]** = Enabled. If enabled, FXS modules answer the call and 'cut through' the voice channel, if there is no other active call on that port, even if the port is in offhooked state. When the call is terminated (by the remote party), the gateway plays a reorder tone for 'TimeForReorderTone' seconds and is then ready to answer the next incoming call, without onhooking the phone. The waiting call is automatically answered by the gateway when the current call is terminated (EnableCallWaiting=1). **Note:** This option is applicable only to FXS modules.
Enable User-Information Usage **[EnableUserInfoUsage]**	Enables or disables usage of the User Information loaded to the gateway via the User Information auxiliary file. 0 = Disable (default). 1 = Enable.
Out-Of-Service Behavior **[FXSOOSBehavior]**	Determines the behavior of FXS endpoints that are not defined (in the Endpoint Phone Number table), and the behavior of all FXS endpoints when a Busy-Out condition exists. 0 (None) = Normal operation: No response is provided to undefined endpoints. Dial tone is played to FXS endpoints when a Busy-Out condition exists. 1 (Reorder Tone) = The gateway plays a reorder tone to the connected phone/PBX (default). 2 (Polarity Reversal) = The gateways reverses the polarity of the endpoint, marking it as unusable (relevant, for example, to PBX DID lines). This option can't be configured on-the-fly. 3 (Polarity Reversal + Reorder Tone) = Same as 2 and 3 combined. This option can't be configured on-the-fly.

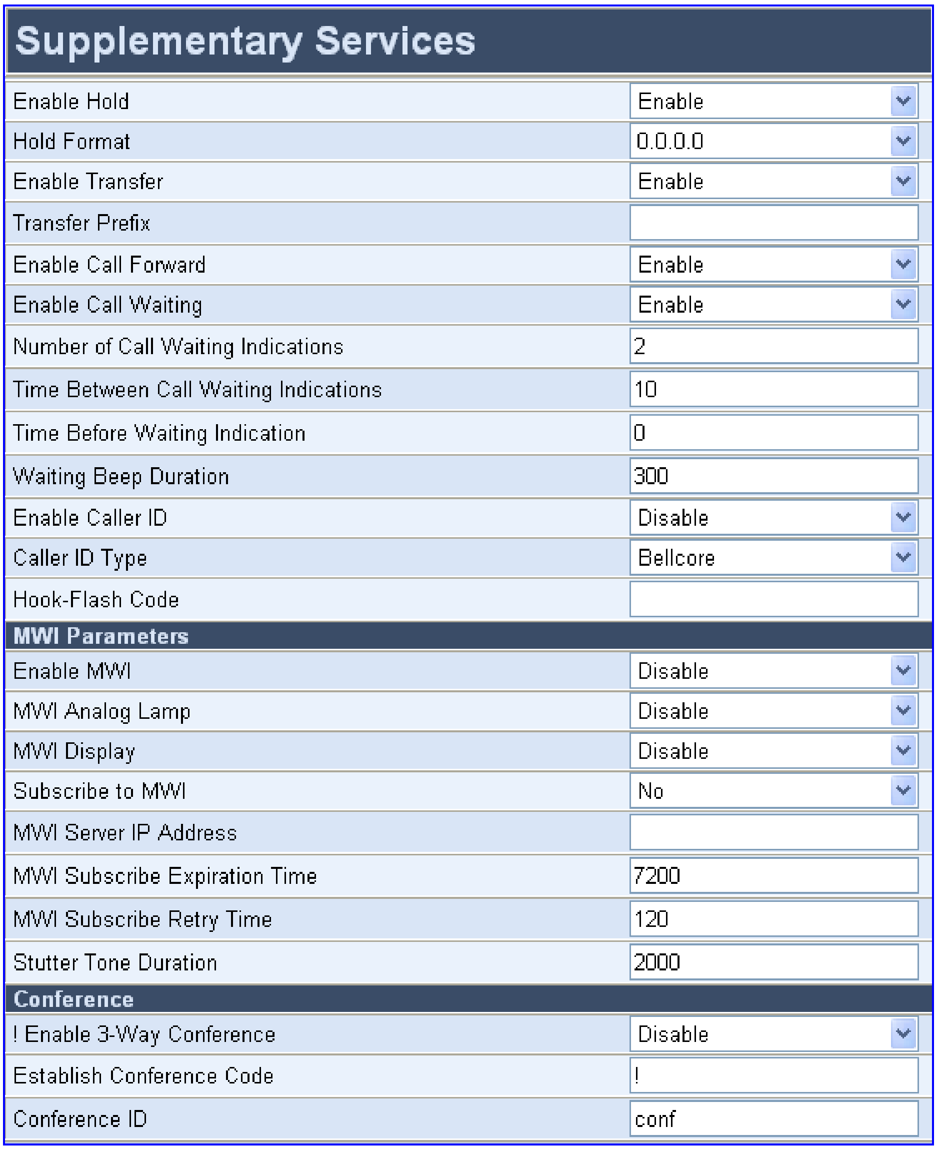
5.5.2.2 Supplementary Services

Use this screen to configure parameters that are associated with supplementary services. For detailed information on the supplementary services, refer to Section 9.1 on page 267.

➢ **To configure the supplementary services' parameters, take these 4 steps:**

1. Open the 'Supplementary Services' screen (**Protocol Management** menu > **Advanced Parameters** submenu > **Supplementary Services** option); the 'Supplementary Services' screen is displayed.

Figure 5-10: Supplementary Services Screen



Supplementary Services

Enable Hold	Enable
Hold Format	0.0.0.0
Enable Transfer	Enable
Transfer Prefix	
Enable Call Forward	Enable
Enable Call Waiting	Enable
Number of Call Waiting Indications	2
Time Between Call Waiting Indications	10
Time Before Waiting Indication	0
Waiting Beep Duration	300
Enable Caller ID	Disable
Caller ID Type	Bellcore
Hook-Flash Code	
MWI Parameters	
Enable MWI	Disable
MWI Analog Lamp	Disable
MWI Display	Disable
Subscribe to MWI	No
MWI Server IP Address	
MWI Subscribe Expiration Time	7200
MWI Subscribe Retry Time	120
Stutter Tone Duration	2000
Conference	
! Enable 3-Way Conference	Disable
Establish Conference Code	!
Conference ID	conf

2. Configure the supplementary services parameters according to Table 5-8.

3. Click the **Submit** button to save your changes, or click the **Subscribe to MWI** or **Un-Subscribe to MWI** buttons to save your changes and to subscribe / unsubscribe to the MWI server.

4. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Table 5-8: Supplementary Services Parameters (continues on pages 92 to 94)

Parameter	Description
Enable Hold **[EnableHold]**	No **[0]** = Disable the Hold service. Yes **[1]** = Enable the Hold service (default). If the Hold service is enabled, a user can activate Hold (or Unhold) using the hook-flash. On receiving a Hold request, the remote party is put on-hold and hears the hold tone. **Note:** To use this service, the gateways at both ends must support this option.
Hold Format **[HoldFormat]**	Determines the format of the hold request. 0.0.0.0 **[0]** = The connection IP address in SDP is 0.0.0.0 (default). Send Only **[1]** = The last attribute of the SDP contains the following 'a=sendonly'.
Enable Transfer **[EnableTransfer]**	No **[0]** = Disable the call transfer service. Yes **[1]** = Enable the call transfer service (using REFER) (default). If the Transfer service is enabled, the user can activate Transfer using hook-flash signaling. If this service is enabled, the remote party performs the call transfer. **Note 1:** To use this service, the gateways at both ends must support this option. **Note 2:** To use this service, set the parameter 'Enable Hold' to 'Yes'.
Transfer Prefix **[xferPrefix]**	Defined string that is added, as a prefix, to the transferred / forwarded called number, when REFER / 3xx message is received. **Note 1:** The number manipulation rules apply to the user part of the 'REFER-TO / Contact' URI before it is sent in the INVITE message. **Note 2:** The 'xferprefix' parameter can be used to apply different manipulation rules to differentiate the transferred / forwarded number from the original dialed number.
Enable Call Forward **[EnableForward]**	No **[0]** = Disable the Call Forward service. Yes **[1]** = Enable Call Forward service (using REFER) (default). For FXS modules a Call Forward table must be defined to use the Call Forward service. To define the Call Forward table, refer to Section 5.5.6.4 on page 124. **Note:** To use this service, the gateways at both ends must support this option.
Enable Call Waiting **[EnableCallWaiting]**	No **[0]** = Disable the Call Waiting service. Yes **[1]** = Enable the Call Waiting service (default). If enabled, when an FXS module receives a call on a busy endpoint, it responds with a 182 response (and not with a 486 busy). The gateway plays a call waiting indication signal. When hook-flash is detected, the gateway switches to the waiting call. The gateway that initiated the waiting call plays a Call Waiting Ringback tone to the calling party after a 182 response is received. **Note 1:** The gateway's Call Progress Tones file must include a 'call waiting Ringback' tone (caller side) and a 'call waiting' tone (called side, FXS only). **Note 2:** The 'Enable Hold' parameter must be enabled on both the calling and the called sides. For information on the Call Waiting feature, refer to Section 9.1.5 on page 269. For information on the Call Progress Tones file, refer to Section 17.1 on page 391.
Number of Call Waiting Indications **[NumberOfWaitingIndications]**	Number of waiting indications that are played to the receiving side of the call (FXS only) for Call Waiting. This parameter is used to control the Registration / Subscription rate. The default value is 2.
Time Between Call Waiting Indications **[TimeBetweenWaitingIndications]**	Difference (in seconds) between call waiting indications (FXS only) for call waiting. The default value is 10 seconds.
Time Before Waiting Indication **[TimeBeforeWaitingIndication]**	Defines the interval (in seconds) before a call waiting indication is played to the port that is currently in a call (FXS only). The valid range is 0 to 100. The default time is 0 seconds.

Table 5-8: Supplementary Services Parameters (continues on pages 92 to 94)

Parameter	Description
Waiting Beep Duration **[WaitingBeepDuration]**	Duration (in msec) of waiting indications that are played to the receiving side of the call (FXS only) for Call Waiting. The default value is 300.
Enable Caller ID **[EnableCallerID]**	No **[0]** = Disable the Caller ID service (default). Yes **[1]** = Enable the Caller ID service. If the Caller ID service is enabled, then, for FXS modules, calling number and Display text are sent to gateway port. For FXO modules, the Caller ID signal is detected and is sent to IP in SIP INVITE message (as 'Display' element). For information on the Caller ID table, refer to Section 5.5.6.3 on page 123. To disable/enable caller ID generation per port, refer to Section 5.5.6.4 on page 124.
Caller ID Type **[CallerIDType]**	Defines one of the following standards for detection (FXO) and generation (FXS) of Caller ID and detection (FXO) of MWI (when specified) signals. Bellcore **[0]** (Caller ID and MWI) (default). ETSI **[1]** (Caller ID and MWI) NTT **[2]** Backward Compatible **[3]** British **[4]** DTMF ETSI **[16]** Denmark **[17]** (Caller ID and MWI) India **[18]** Brazil **[19]** **Note 1:** The Caller ID signals are generated/detected between the first and the second rings. **Note 2:** To select the Bellcore Caller ID sub standard, use the parameter 'BellcoreCallerIDTypeOneSubStandard'. To select the ETSI Caller ID sub standard, use the parameter 'ETSICallerIDTypeOneSubStandard'. **Note 3:** To select the Bellcore MWI sub standard, use the parameter 'BellcoreVMWITypeOneStandard'. To select the ETSI MWI sub standard, use the parameter 'ETSIVMWITypeOneStandard'.
Hook-Flash Code **[HookFlashCode]**	Determines the digit pattern used by the PBX to indicate a 'Hook-Flash' event. When this pattern is detected from the Tel side, the gateway responds as if a Hook-Flash event occurs and sends an INFO message indicating 'Hook Flash'. If configured and a Hook-Flash indication is received from the IP side, the gateway generates this pattern to the Tel side. The valid range is a 25-character string.
MWI Parameters	
Enable MWI **[EnableMWI]**	Enable MWI (message waiting indication). Disable **[0]** = Disabled (default). Enable **[1]** = MWI service is enabled. This parameter is applicable only to FXS modules. **Note:** The Mediant 1000 only supports reception of MWI. For detailed information on MWI, refer to Section 9.1.6 on page 270.
MWI Analog Lamp **[MWIAnalogLamp]**	Disable **[0]** = Disable (default). Enable **[1]** = Enable visual Message Waiting Indication, supplies line voltage of approximately 100 VDC to activate the phone's lamp. This parameter is applicable only to FXS modules.
MWI Display **[MWIDisplay]**	Disable **[0]** = MWI information isn't sent to display (default). Enable **[1]** = MWI information is sent to display. If enabled, the gateway generates an MWI FSK message that is displayed on the MWI display. This parameter is applicable only to FXS modules.

Table 5-8: Supplementary Services Parameters (continues on pages 92 to 94)

Parameter	Description
Subscribe to MWI **[EnableMWISubscription]**	Disable **[0]** = Disable MWI subscription (default). Enable **[1]** = Enable subscription to MWI (to MWIServerIP address). **Note:** Use the parameter 'SubscriptionMode' (described in Table 5-4 on page 72) to determine whether the gateway subscribes separately per endpoint of for the entire gateway.
MWI Server IP Address **[MWIServerIP]**	MWI server IP address. If provided, the gateway subscribes to this IP address. Can be configured as a numerical IP address or as a domain name. If not configured, the Proxy IP address is used instead.
MWI Subscribe Expiration Time **[MWIExpirationTime]**	MWI subscription expiration time in seconds. The default is 7200 seconds. The range is 10 to 72000.
MWI Subscribe Retry Time **[SubscribeRetryTime]**	Subscription retry time in seconds. The default is 120 seconds. The range is 10 to 7200.
Stutter Tone Duration **[StutterToneDuration]**	Duration (in msec) of the played Stutter dial tone, which indicates that Call Forwarding is enabled or that there is a waiting message(s). The default is 2,000 (i.e., 2 seconds). The range is 1,000 to 60,000. The Stutter tone is played (instead of a regular Dial tone), when a Call Forward is enabled on the specific port or when MWI is received. The tone is composed of a 'Confirmation tone', which is played for a user-defined duration (StutterToneDuration), followed by a 'Stutter tone'. Both tones are defined in the CPT file. **Note 1:** This parameter is applicable only to FXS gateways. **Note 2:** The message waiting notification (MWI) tone takes precedence over the call forwarding reminder tone. For detailed information on Message Waiting Indication (MWI), refer to Section 9.1.6 on page 270.
Conference Parameters	
Enable 3-Way Conference **[Enable3WayConference]**	Enables or disables the 3-Way Conference feature. Valid options include: 0 = Disable (default) 1 = Enable
Establish Conference Code **[ConferenceCode]**	Defines the digit pattern that once detected, generates the Conference-initiating INVITE when Enable3WayConference is set to 1. The valid range is a 25-character string. The default is "!" (Hook-Flash).
Conference ID **[ConferenceID]**	Defines the Conference Identification string (up to 16 characters). ▪ For 3-way conferencing using an external media server: The gateway uses this identifier in the Conference-initiating INVITE that is sent to the media server when Enable3WayConferenceis set to 1. ▪ When using the Mediant 1000 Conference module: To join a conference, the INVITE URI must include the Conference ID string, preceded by the number of the participants in the conference, and terminated by a unique number. For example: INVITE sip:4MyConference1234@10.1.10.10. INVITE messages with the same URI join the same conference. The default value is 'conf'. For example: ConferenceID = MyConference.
3WayConferenceMode	For a description of this *ini* file parameter, refer to Section 6.5.7 on page 230.
BipOnConference	For a description of this *ini* file parameter, refer to Section 6.5.7 on page 230.

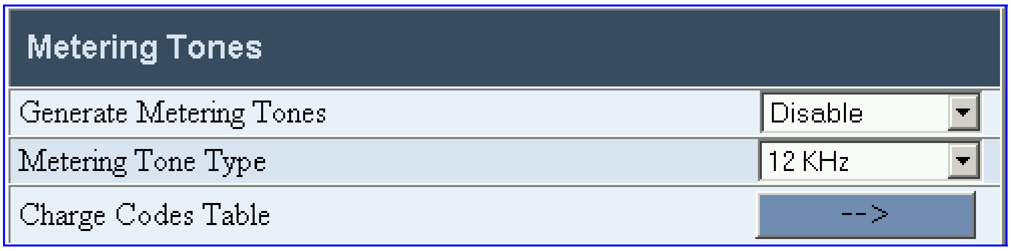
5.5.2.3 Metering Tones

FXS modules can generate 12/16 KHz metering pulses towards the Tel side (e.g., for connection to a payphone or private meter). Tariff pulse rate is determined according to an internal table. This capability enables users to define different tariffs according to the Source / Destination numbers and the Time-of-Day. The tariff rate includes the time interval between the generated pulses and the number of pulses generated on answer.

➢ To configure the Metering Tones, take these 6 steps:

1. Open the 'Metering Tones' screen (**Protocol Management** menu > **Advanced Parameters** submenu > **Metering Tones** option); the 'Metering Tones' screen is displayed.

Figure 5-11: Metering Tones Parameters Screen



Metering Tones

Generate Metering Tones	Disable
Metering Tone Type	12 KHz
Charge Codes Table	-->

2. In the field 'Metering Tone Type', select the type of the metering tone according to your requirements (refer to Table 5-9).
3. In the field 'Generate Metering Tones', select the method used to configure the metering tones that are generated to the Tel side (refer to Table 5-9).
   - If you selected 'Internal Table' in the field 'Generate Metering Tones', you must configure the 'Charge Codes Table'. To configure the 'Charge Codes Table', refer to Section 5.5.2.3.1 below. Continue with step 4.
   - The 'RADIUS' option is not supported in the current version.
4. In the Tel to IP Routing table (Section 5.5.4.2 on page 107) assign a charge code rule to the routing rules you require.
   When a new call is established, the Tel to IP Routing table is searched for the destination IP addresses. Once a route is found, the Charge Code (configured for that route) is used to associate the route with an entry in the Charge Codes table.
5. Click the **Submit** button to save your changes.
6. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Table 5-9: Metering Tones Parameters

Parameter	Description
Generate Metering Tones **[PayPhoneMeteringMode]**	Determines the method used to configure the metering tones that are generated to the Tel side (FXS modules only). Disable **[0]** = Metering tones aren't generated (default). Internal Table **[1]** = Metering tones are generated according to the internal table configured by the parameter ChargeCode. RADIUS **[2]** = N/A. **Note:** This parameter is not applicable to the Metering Tones Relay mechanism (described in Section 13.1 on page 339).

Table 5-9: Metering Tones Parameters

Parameter	Description
Metering Tone Type **[MeteringType]**	Defines the metering tone (12 kHz or 16 kHz) that is detected by FXO modules and generated by FXS modules. 12 kHz **[0]** = 12 kHz metering tone (default). 16 kHz **[1]** = 16 kHz metering tone. **Note:** Suitable (12 kHz or 16 KHz) *coeff* must be used for both FXS and FXO modules.
Charge Codes Table	For detailed information on configuring the Charge Codes Table, refer to Section 5.5.2.3.1 below.

5.5.2.3.1 Charge Codes Table

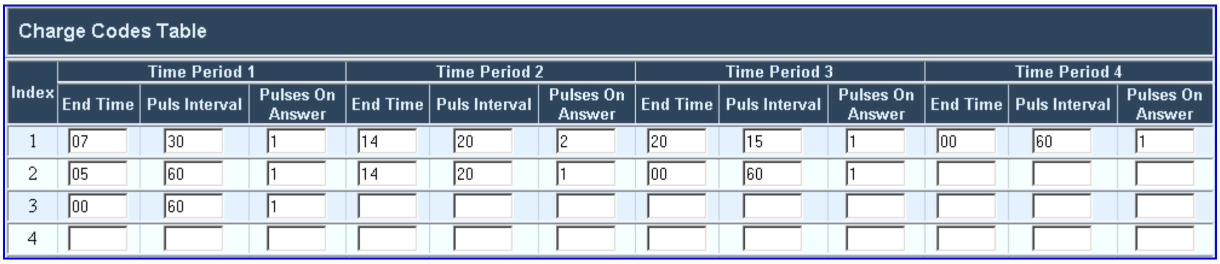
The Charge Codes table is used to configure the metering tones (and their time interval) that the FXS module generates to the Tel side. To associate a charge code to an outgoing Tel to IP call, use the Tel to IP Routing table.

You can also configure the Charge Codes table using the *ini* file parameter ChargeCode (refer to Section 6.5.10 on page 242).

➢ To configure the Charge Codes table, take these 6 steps:

1. Access the 'Metering Tones' screen (**Protocol Management** menu > **Advanced Parameters** submenu > **Metering Tones** option); the 'Metering Tones' screen is displayed (Figure 5-11).
2. Open the 'Charge Codes Table' screen by clicking the arrow sign (-->) to the right of the Charge Codes Table label; the Charge Codes Table is displayed.

Figure 5-12: Charge Codes Table Screen



Charge Codes Table

Index	Time Period 1			Time Period 2			Time Period 3			Time Period 4		
	End Time	Puls Interval	Pulses On Answer	End Time	Puls Interval	Pulses On Answer	End Time	Puls Interval	Pulses On Answer	End Time	Puls Interval	Pulses On Answer
1	07	30	1	14	20	2	20	15	1	00	60	1
2	05	60	1	14	20	1	00	60	1			
3	00	60	1									
4												

3. Use the table to define up to 25 different charge codes (each charge code is defined in a single row). Each code can include from a single and up to four different time periods in a day (24 hours). Each time period is composed of:
   - The end (in a 24 rounded-hour's format) of the time period.
   - The time interval between pulses (in seconds).
   - The number of pulses sent on answer.
4. The first time period always starts at midnight (00). It is mandatory that the last time period of each rule ends at midnight (00). This prevents undefined time frames in a day.
   The gateway selects the time period by comparing the gateway's current time to the end time of each time period of the selected Charge Code. The gateway generates the Number of Pulses on Answer once the call is connected and from that point on, it generates a pulse each Pulse Interval. If a call starts at a certain time period and crosses to the next, the information of the next time period is used.
5. Click the **Submit** button to save your changes.
6. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

5.5.2.4 Keypad Features

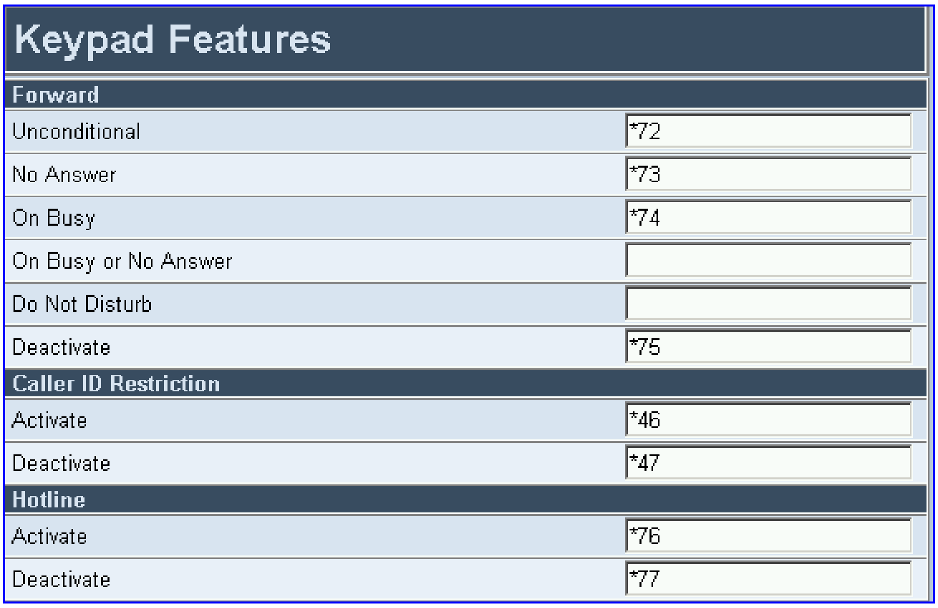
The Keypad Features screen (applicable only to FXS modules) enables you to activate and deactivate the following features directly from the connected telephone's keypad:

- Hotline (refer to Section 5.5.6.2 on page 122).
- Caller ID Restriction (refer to Section 5.5.6.3 on page 123).
- Call Forward (refer to Section 5.5.6.4 on page 124).

➢ To configure the keypad features, take these 4 steps:

1. Open the 'Keypad Features' screen (**Protocol Management** menu > **Advanced Parameters** submenu > **Keypad Features** option); the 'Keypad Features' screen is displayed.

Figure 5-13: Keypad Features Screen



Keypad Features

Forward	
Unconditional	*72
No Answer	*73
On Busy	*74
On Busy or No Answer	
Do Not Disturb	
Deactivate	*75
Caller ID Restriction	
Activate	*46
Deactivate	*47
Hotline	
Activate	*76
Deactivate	*77

2. Configure the Keypad Features according to Table 5-10.
3. Click the **Submit** button to save your changes.
4. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Notes:

- The method used by the gateway to collect dialed numbers is identical to the method used during a regular call (i.e., max digits, interdigit timeout, digit map, etc.).
- The activation of each feature remains in effect until it is deactivated (i.e., it is not per call).

Table 5-10: Keypad Features Parameters (continues on pages 98 to 98)

Parameter	Description
Forward Note that the forward type and number can be viewed in the Call Forward Table (refer to Section 5.5.6.4 on page 124)	
Unconditional **[KeyCFUnCond]**	Keypad sequence that activates the immediate forward option.
No Answer **[KeyCFNoAnswer]**	Keypad sequence that activates the forward on no answer option.
On Busy **[KeyCFBusy]**	Keypad sequence that activates the forward on busy option.
On Busy or No Answer **[KeyCFBusyOrNoAnswer]**	Keypad sequence that activates the forward on 'busy or no answer' option.
Do Not Disturb **[KeyCFDoNotDisturb]**	Keypad sequence that activates the Do Not Disturb option (immediately reject incoming calls).
To activate the required forward method from the telephone: ▪ Dial the preconfigured sequence number on the keypad; a dial tone is heard. ▪ Dial the telephone number to which the call is forwarded (terminate the number with #); a confirmation tone is heard.	
Deactivate **[KeyCFDeact]**	Keypad sequence that deactivates any of the forward options. After the sequence is pressed a confirmation tone is heard.
Caller ID Restriction Note that the caller ID presentation can be viewed in the Caller Display Information table (refer to Section 5.5.6.3 on page 123)	
Activate **[KeyCLIR]**	Keypad sequence that activates the restricted Caller ID option. After the sequence is pressed a confirmation tone is heard.
Deactivate **[KeyCLIRDeact]**	Keypad sequence that deactivates the restricted Caller ID option. After the sequence is pressed a confirmation tone is heard.
Hotline Note that the destination phone number and the auto dial status can be viewed in the Automatic Dialing table (refer to Section 5.5.6.2 on page 122)	
Activate **[KeyHotLine]**	Keypad sequence that activates the delayed hotline option. To activate the delayed hotline option from the telephone: ▪ Dial the preconfigured sequence number on the keypad; a dial tone is heard. ▪ Dial the telephone number to which the phone automatically dials after a configurable delay (terminate the number with #); a confirmation tone is heard.
Deactivate **[KeyHotLineDeact]**	Keypad sequence that deactivates the delayed hotline option. After the sequence is pressed a confirmation tone is heard.

5.5.3 Configuring the Number Manipulation Tables

The VoIP gateway provides four Number Manipulation tables for incoming and outgoing calls. These tables are used to modify the destination and source telephone numbers so that the calls can be routed correctly.

The Manipulation Tables are:

- Destination Phone Number Manipulation Table for IP→Tel calls
- Destination Phone Number Manipulation Table for Tel→IP call
- Source Phone Number Manipulation Table for IP→Tel calls
- Source Phone Number Manipulation Table for Tel→IP calls

Note: Number manipulation can occur either before or after a routing decision is made. For example, you can route a call to a specific hunt (analog module) or trunk (digital module) group according to its original number, and then you can remove / add a prefix to that number before it is routed. To control when number manipulation is done, define the IP to Tel Routing Mode (described in Table 5-16) and Tel to IP Routing Mode (described in Table 5-15) parameters.

Possible uses for number manipulation can be as follows:

- To strip/add dialing plan digits from/to the number. For example, a user could dial 9 in front of each number in order to indicate an external line. This number (9) can be removed here before the call is setup.
- Allow / disallow Caller ID information to be sent according to destination / source prefixes. For detailed information on Caller ID, refer to Section 5.5.6.3 on page 123.
- **For digital modules only:** Assignment of NPI/TON to IP→Tel calls. The VoIP gateway can use a single global setting for NPI/TON classification or it can use the setting in this table on a call by call basis. Control for this is done using ‘Protocol Management>Protocol Definition>Destination/Source Number Encoding Type’.

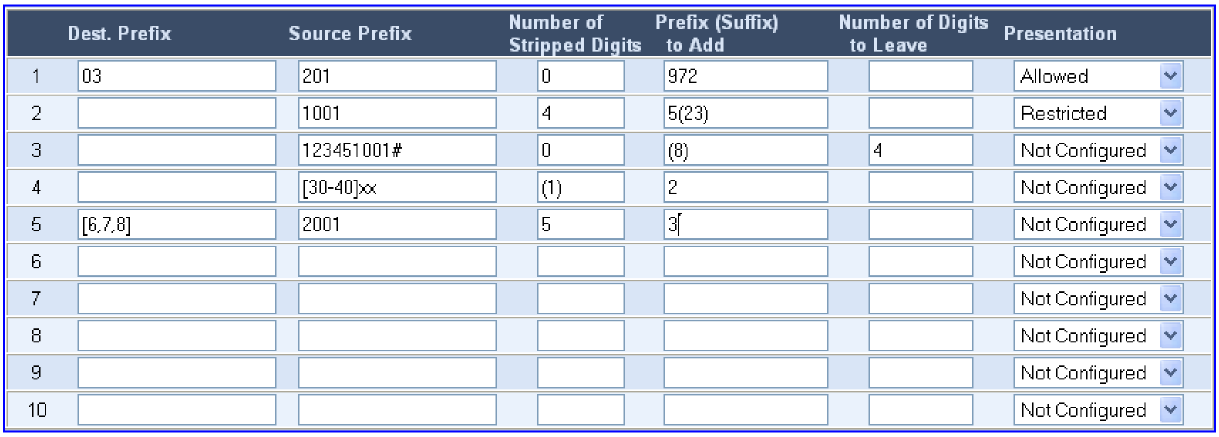
For configuring the Number Manipulation Tables, you can also use the following *ini* file parameters (refer to Section 6.5.10 on page 242):

- NumberMapTel2IP: configures the Destination Phone Number Manipulation Table for Tel to IP Calls
- NumberMapIP2Tel: configures the Destination Phone Number Manipulation Table for IP to Tel Calls
- SourceNumberMapTel2IP: configures the Source Phone Number Manipulation Table for IP to Tel Calls
- SourceNumberMapIP2Tel: configures the Source Phone Number Manipulation Table for Tel to IP Calls

➢ To configure the Number Manipulation tables, take these 5 steps:

1. Open the Number Manipulation screen you want to configure (**Protocol Management** menu > **Manipulation Tables** submenu); the relevant Manipulation table screen is displayed. Figure 5-14 shows the 'Source Phone Number Manipulation Table for Tel→IP calls'.

Figure 5-14: Source Phone Number Manipulation Table for Tel→IP calls



	Dest. Prefix	Source Prefix	Number of Stripped Digits	Prefix (Suffix) to Add	Number of Digits to Leave	Presentation
1	03	201	0	972		Allowed
2		1001	4	5(23)		Restricted
3		123451001#	0	(8)	4	Not Configured
4		[30-40]xx	(1)	2		Not Configured
5	[6,7,8]	2001	5	3		Not Configured
6						Not Configured
7						Not Configured
8						Not Configured
9						Not Configured
10						Not Configured

2. In the 'Table Index' drop-down list, select the range of entries that you want to edit (up to 20 entries can be configured for Source Number Manipulation and 50 entries for Destination Number Manipulation).
3. Configure the Number Manipulation table according to Table 5-11.
4. Click the **Submit** button to save your changes.
5. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Table 5-11: Number Manipulation Parameters (continues on pages 100 to 101)

Parameter	Description
Destination Prefix	Each entry in the Destination Prefix fields represents a destination telephone number prefix. An asterisk (*) represents any number.
Source Prefix	Each entry in the Source Prefix fields represents a source telephone number prefix. An asterisk (*) represents any number.
Source IP	Each entry in the Source IP fields represents the source IP address of the call (obtained from the Contact header in the INVITE message). This column only applies to the 'Destination Phone Number Manipulation Table for IP to Tel'. **Note:** The source IP address can include the 'x' wildcard to represent single digits. For example: 10.8.8.xx represents all the addresses between 10.8.8.10 to 10.8.8.99.
The manipulation rules are applied to any incoming call whose: ▪ Destination number prefix matches the prefix defined in the 'Destination Number' field. ▪ Source number prefix matches the prefix defined in the 'Source Prefix' field. ▪ Source IP address matches the IP address defined in the 'Source IP' field (if applicable). Note that number manipulation can be performed using a combination of each of the above criteria, or using each criterion independently. **Note:** For available notations that represent multiple numbers, refer to Section 5.5.3.1 on page 102.	

Table 5-11: Number Manipulation Parameters (continues on pages 100 to 101)

Parameter	Description
Number of Stripped Digits	▪ Enter the number of digits that you want to remove from the left of the telephone number prefix. For example, if you enter 3 and the phone number is 5551234, the new phone number is 1234. ▪ Enter the number of digits (in brackets) that you want to remove from the right of the telephone number prefix. **Note:** A combination of the two options is allowed (e.g., 2(3)).
Prefix (Suffix) to Add	▪ Prefix - Enter the number / string you want to add to the front of the phone number. For example, if you enter 9 and the phone number is 1234, the new number is 91234. ▪ Suffix - Enter the number / string (in brackets) you want to add to the end of the phone number. For example, if you enter (00) and the phone number is 1234, the new number is 123400. **Note:** You can enter a prefix and a suffix in the same field (e.g., 9(00)).
Number of Digits to Leave	Enter the number of digits that you want to leave from the right.
Note: The manipulation rules are executed in the following order: **1.** Num of stripped digits **2.** Number of digits to leave **3.** Prefix / suffix to add Figure 5-14 on the previous page exemplifies the use of these manipulation rules in the 'Source Phone Number Manipulation Table for Tel→IP Calls': ▪ When destination number equals 035000 and source number equals 20155, the source number is changed to 97220155. ▪ When source number equals 1001876, it is changed to 587623. ▪ Source number 1234510012001 is changed to 20018. ▪ Source number 3122 is changed to 2312.	
Presentation	Select 'Allowed' to send Caller ID information when a call is made using these destination / source prefixes. Select 'Restricted' if you want to restrict Caller ID information for these prefixes. When set to 'Not Configured', the privacy is determined according to the Caller ID table (refer to Section 5.5.6.3 on page 123). **Note:** If 'Presentation' is set to 'Restricted' and 'Asserted Identity Mode' is set to 'P-Asserted', the From header in INVITE message is: From: 'anonymous' <sip: anonymous@anonymous.invalid> and 'privacy: id' header is included in the INVITE message.
NPI	Select the Number Plan assigned to this entry. You can select Unknown **[0]**, Private **[9]** or E.164 Public **[1]**. The default is Unknown. For a detailed list of the available NPI/TON values, refer to Section 5.5.3.2 on page 102.
TON	Select the Number Type assigned to this entry. ▪ If you selected Unknown as the Number Plan, you can select Unknown **[0]**. ▪ If you selected Private as the Number Plan, you can select Unknown **[0]**, Level 2 Regional **[1]**, Level 1 Regional **[2]**, PISN Specific **[3]** or Level 0 Regional (Local) **[4]**. ▪ If you selected E.164 Public as the Number Plan, you can select Unknown **[0]**, International **[1]**, National **[2]**, Network Specific **[3]**, Subscriber **[4]** or Abbreviated **[6]**. The default is Unknown.

5.5.3.1 Dialing Plan Notation

The dialing plan notation applies, in addition to the four Manipulation tables, also to Tel→IP Routing table and to IP→Trunk Group Routing table.

When entering a number in the destination and source 'Prefix' columns, you can create an entry that represents multiple numbers using the following notation:

- [n-m] represents a range of numbers
- [n,m] represents multiple numbers. Note that this notation only supports single digit numbers.
- x represents any single digit
- # (that terminates the number) represents the end of a number
- A single asterisk (*) represents any number

For example:

- [5551200-5551300]# represents all numbers from 5551200 to 5551300
- [2,3,4]xxx# represents four-digit numbers that start with 2, 3 or 4
- 54324 represents any number that starts with 54324
- 54324xx# represents a 7 digit number that starts with 54324
- 123[100-200]# represents all numbers from 123100 to 123200.

The VoIP gateway matches the rules starting at the top of the table (i.e. top rules take precedence over lower rules). For this reason, enter more specific rules above more generic rules. For example, if you enter 551 in entry 1 and 55 in entry 2, the VoIP gateway applies rule 1 to numbers that starts with 551 and applies rule 2 to numbers that start with 550, 552, 553, 554, 555, 556, 557, 558 and 559. However if you enter 55 in entry 1 and 551 in entry 2, the VoIP gateway applies rule 1 to all numbers that start with 55 including numbers that start with 551.

5.5.3.2 Numbering Plans and Type of Number

Numbers are classified by their Numbering Plan Indication (NPI) and their Type of Number (TON). The Mediant 1000 supports all NPI/TON classifications used in the standard. The list of ISDN ETSI NPI/TON values is shown as follows:

Table 5-12: NPI/TON Values for ISDN ETSI

NPI	TON	Description
Unknown [0]	Unknown [0]	A valid classification, but one that has no information about the numbering plan.
E.164 Public [1]	Unknown [0]	A public number in E.164 format, but no information on what kind of E.164 number.
	International [1]	A public number in complete international E.164 format. For example: 16135551234
	National [2]	A public number in complete national E.164 format. For example: 6135551234
	Subscriber [4]	A public number in complete E.164 format representing a local subscriber. For example: 5551234
Private [9]	Unknown [0]	A private number, but with no further information about the numbering plan
	Level 2 Regional [1]	
	Level 1 Regional [2]	A private number with a location. For example: 3932200
	PISN Specific [3]	
	Level 0 Regional (local) [4]	A private local extension number. For example: 2200

For NI-2 and DMS-100 ISDN variants the valid combinations of TON and NPI for calling and called numbers are (Plan/Type):

- 0/0 - Unknown/Unknown
- 1/1 - International number in ISDN/Telephony numbering plan
- 1/2 - National number in ISDN/Telephony numbering plan
- 1/4 - Subscriber (local) number in ISDN/Telephony numbering plan
- 9/4 - Subscriber (local) number in Private numbering plan

5.5.3.3 Mapping NPI/TON to Phone-Context

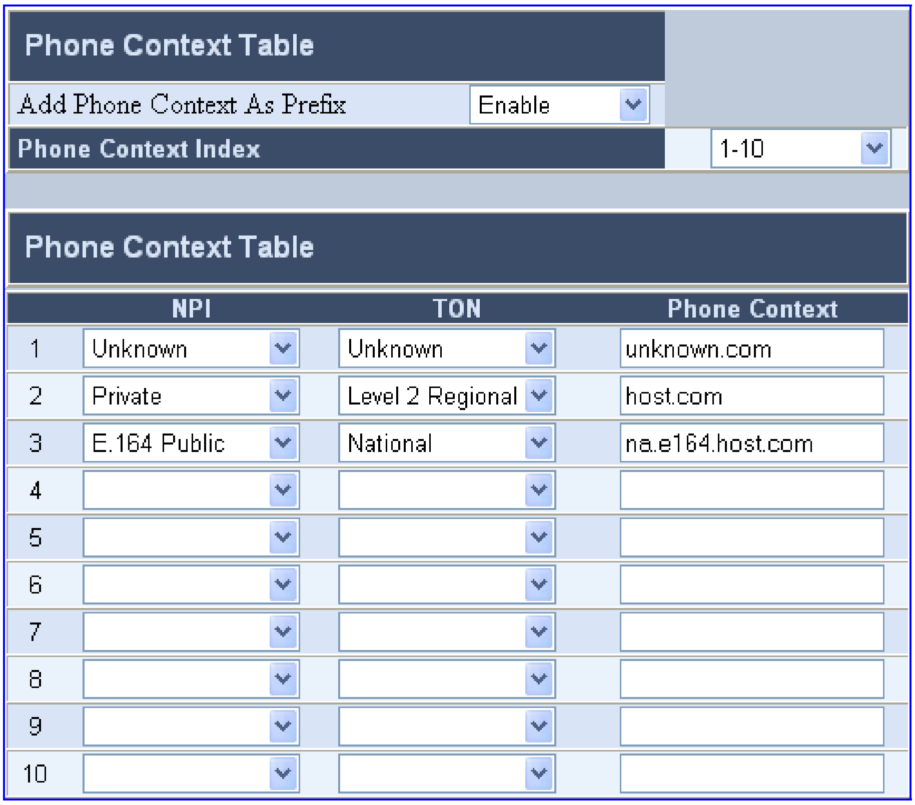
The Phone-Context table is used to configure the mapping of NPI and TON to the Phone-Context SIP parameter. When a call is received from the ISDN, the NPI and TON are compared against the table and the Phone-Context value is used in the outgoing SIP INVITE message. The same mapping occurs when an INVITE with a Phone-Context attribute is received. The Phone-Context parameter appears in the standard SIP headers where a phone number is used (Request-URI, To, From, Diversion).

You can also configure the Phone Context table using the *ini* file parameter PhoneContext (refer to Table 6-11 on page 247).

➢ To configure the Phone-Context tables, take these 6 steps:

1. Open the 'Phone Context Table' screen (**Protocol Management** menu > **Manipulation Tables** submenu > **Phone Context Table** option); the 'Phone Context Table' screen appears, as shown below.

Figure 5-15: Phone Context Table Screen



Phone Context Table

Add Phone Context As Prefix: Enable

Phone Context Index: 1-10

Phone Context Table

	NPI	TON	Phone Context
1	Unknown	Unknown	unknown.com
2	Private	Level 2 Regional	host.com
3	E.164 Public	National	na.e164.host.com
4			
5			
6			
7			
8			
9			
10			

2. From the ‘Add Phone Context As Prefix' drop-down list, select 'Enable' to add the received Phone-Context parameter as a prefix to outgoing ISDN SETUP Called and Calling numbers, if necessary.

3. From the 'Phone Context Index' drop-down list, select the index number.
4. Configure the Phone Context table according to Table 5-13.
5. Click the **Submit** button to save your changes.
6. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Notes:

- Several rows with the same NPI-TON or Phone-Context are allowed. In such a scenario, a Tel-to-IP call uses the first match.
- Phone-Context '+' is a unique case as it doesn't appear in the Request-URI as a Phone-Context parameter. Instead, it's added as a prefix to the phone number. The '+' isn't removed from the phone number in the IP-to-Tel direction.

Table 5-13: Phone-Context Parameters

Parameter	Description
Add Phone Context As Prefix **[AddPhoneContextAsPrefix]**	Determines whether or not the received Phone-Context parameter is added as a prefix to the outgoing ISDN SETUP Called and Calling numbers. Valid options include: 0 = Disable (default). 1 = Enable.
NPI	Select the Number Plan assigned to this entry. You can select the following: 0 = Unknown (default) 1 = E.164 Public 9 = Private For a detailed list of the available NPI/TON values, refer to Section 5.5.3.2 on page 102.
TON	Select the Number Type assigned to this entry. ▪ If you selected Unknown as the NPI, you can select Unknown (0) ▪ If you selected Private as the NPI, you can select Unknown (0), Level 2 Regional (1), Level 1 Regional (2), PSTN Specific (3), or Level 0 Regional (Local) (4). ▪ If you selected E.164 Public as the NPI, you can select Unknown (0), International (1), National (2), Network Specific (3), Subscriber (4), or Abbreviated (6).
Phone Context	The Phone-Context SIP URI parameter.

5.5.4 Configuring the Routing Tables

Use this submenu to configure the gateway's IP→Tel and Tel→IP routing tables and their associated parameters:

- General Parameters (refer to Section 5.5.4.1 on page 105)
- Tel to IP Routing Table (refer to Section 5.5.4.2 on page 107)
- IP to Trunk Group Routing (refer to Section 5.5.4.3 on page 109)
- Internal DNS Table (refer to Section 5.5.4.4 on page 111)
- Internal SRV Table (refer to Section 5.5.4.5 on page 112)
- Reasons for Alternative Routing (refer to Section 5.5.4.6 on page 113)
- Release Cause Mapping (refer to Section 5.5.4.7 on page 114)

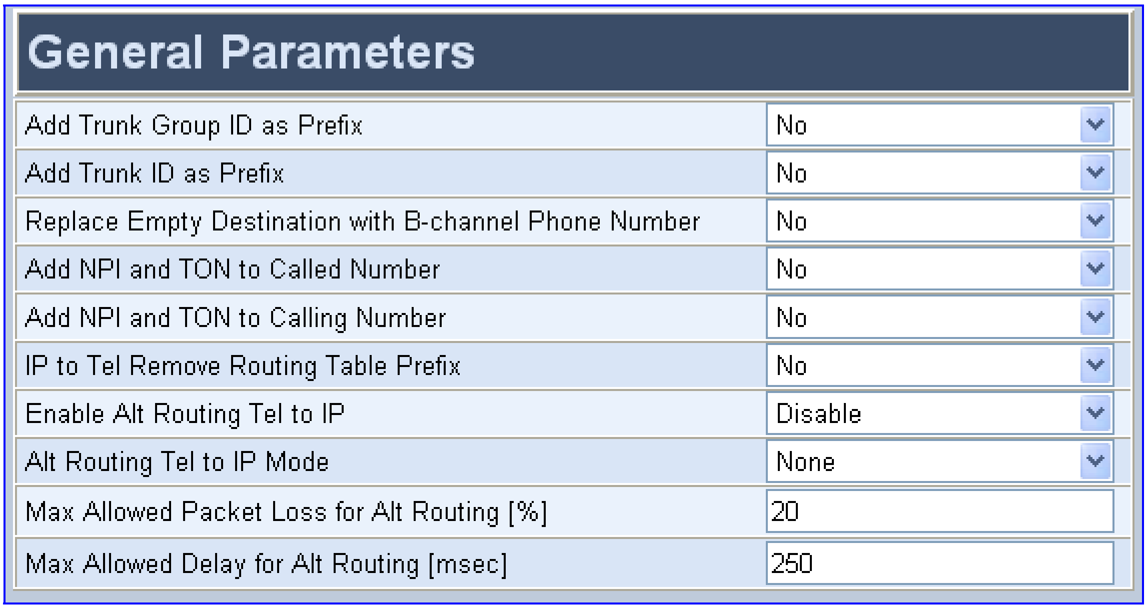
5.5.4.1 General Parameters

Use this screen to configure the gateway's IP→Tel and Tel→IP routing parameters.

➢ **To configure the general parameters under Routing Tables, take these 4 steps:**

1. Open the 'General Parameters' screen (**Protocol Management** menu > **Routing Tables** submenu > **General Parameters** option); the 'General Parameters' screen is displayed.

Figure 5-16: Routing Tables, General Parameters Screen



General Parameters

Add Trunk Group ID as Prefix	No
Add Trunk ID as Prefix	No
Replace Empty Destination with B-channel Phone Number	No
Add NPI and TON to Called Number	No
Add NPI and TON to Calling Number	No
IP to Tel Remove Routing Table Prefix	No
Enable Alt Routing Tel to IP	Disable
Alt Routing Tel to IP Mode	None
Max Allowed Packet Loss for Alt Routing [%]	20
Max Allowed Delay for Alt Routing [msec]	250

2. Configure the general parameters under 'Routing Tables' according to Table 5-14.
3. Click the **Submit** button to save your changes.
4. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Table 5-14: Routing Tables, General Parameters (continues on pages 105 to 106)

Parameter	Description
Add Trunk Group ID as Prefix **[AddTrunkGroupAsPrefix]**	No **[0]** = Don't add trunk group ID as prefix (default). Yes **[1]** = Add trunk group ID as prefix to called number. If enabled, then the trunk group ID is added as a prefix to the destination phone number for Tel→IP calls. **Note 1:** This option can be used to define various routing rules. **Note 2:** To use this feature you must configure the trunk group IDs.
Add Trunk ID as Prefix **[AddPortAsPrefix]**	0 = Don't add (default) 1 = Add trunk ID number (single digit in the range 1 to 8) as a prefix to the called phone number for Tel→IP incoming calls. This option can be used to define various routing rules.
Replace Empty Destination with B-channel Phone Number **[ReplaceEmptyDstWithPortNumber]**	0 = Disabled (default). 1 = Enabled, Internal channel number is used as a destination number if called number is missing. **Note:** Applicable only to Tel→IP calls, if called number is missing.

Table 5-14: Routing Tables, General Parameters (continues on pages 105 to 106)

Parameter	Description
Add NPI and TON to Calling Number **[AddNPIandTON2CallingNumber]**	0 = Do not change the Calling Number (default). 1 = Add NPI and TON to the Calling Number of incoming (Tel to IP) ISDN call. For example: After receiving a Calling Number = 555, NPI = 1 and TON = 3, the modified number is going to be 13555. This number can later be used for manipulation and routing purposes.
Add NPI and TON to Called Number **[AddNPIandTON2CalledNumber]**	0 = Do not change the Called Number (default). 1 = Add NPI and TON to the Called Number of incoming (Tel to IP) ISDN call. For example: After receiving a Called Number = 555, NPI=1 and TON = 3, the modified number is going to be 13555. This number can later be used for manipulation and routing purposes.
IP to Tel Remove Routing Table Prefix **[RemovePrefix]**	No **[0]** = Don't remove prefix (default) Yes **[1]** = Remove the prefix (defined in the IP to Trunk Group Routing table) from a telephone number for an IP→Tel call, before forwarding it to Tel. For example: To route an incoming IP→Tel Call with destination number 21100, the IP to Trunk Group Routing table is scanned for a matching prefix. If such prefix is found, 21 for instance, then before the call is routed to the corresponding trunk group the prefix (21) is removed from the original number, so that only 100 is left. **Note 1:** Applicable only if number manipulation is performed after call routing for IP→Tel calls (refer to 'IP to Tel Routing Mode' parameter -- RouteModeIP2Tel = 0). **Note 2:** Similar operation (of removing the prefix) is also achieved by using the usual number manipulation rules.
Enable Alt Routing Tel to IP **[AltRoutingTel2IPEnable]**	Disable **[0]** = Disable the Alternative Routing feature (default). Enable **[1]** = Enable the Alternative Routing feature. Status Only **[2]** = The Alternative Routing feature is disabled. A read-only information on the quality of service of the destination IP addresses is provided. For information on the Alternative Routing feature, refer to Section 9.8 on page 284.
Alt Routing Tel to IP Mode **[AltRoutingTel2IPMode]**	None **[0]** = Alternative routing is not used. Connectivity **[1]** = Alternative routing is performed if ping to initial destination failed. QoS **[2]** = Alternative routing is performed if poor quality of service was detected. Both **[3]** = Alternative routing is performed if, either ping to initial destination failed, or poor quality of service was detected, or DNS host name is not resolved (default). **Note:** QoS (Quality of Service) is quantified according to delay and packet loss, calculated according to previous calls. QoS statistics are reset if no new data is received for two minutes. For information on the Alternative Routing feature, refer to 9.8 on page 284.
Max Allowed Packet Loss for Alt Routing [%] **[IPConnQoSMaxAllowedPL]**	Packet loss percentage at which the IP connection is considered a failure. The range is 1% to 20%. The default value is 20%.
Max Allowed Delay for Alt Routing [msec] **[IPConnQoSMaxAllowedDelay]**	Transmission delay (in msec) at which the IP connection is considered a failure. The range is 100 to 1000. The default value is 250 msec.

5.5.4.2 Tel to IP Routing Table

The Tel to IP Routing Table is used to route incoming Tel calls to IP addresses. This routing table associates a called / calling telephone number's prefixes with a destination IP address or with an FQDN (Fully Qualified Domain Name). When a call is routed through the VoIP gateway (Proxy isn't used), the called and calling numbers are compared to the list of prefixes on the IP Routing Table (up to 50 prefixes can be configured); Calls that match these prefixes are sent to the corresponding IP address. If the number dialed does not match these prefixes, the call is not made.

When using a Proxy server, you do not need to configure the Tel to IP Routing Table. However, if you want to use fallback routing when communication with Proxy servers is lost, or to use the 'Filter Calls to IP' and 'IP Security' features, or to obtain different SIP URI host names (per called number) or to assign IP profiles, you need to configure the IP Routing Table.

Note that for the Tel to IP Routing table to take precedence over a Proxy for routing calls, set the parameter 'PreferRouteTable' to 1. The gateway checks the 'Destination IP Address' field in the 'Tel to IP Routing' table for a match with the outgoing call. Only if a match is not found, a Proxy is used.

Possible uses for Tel to IP Routing include the following:

- Can fallback to internal routing table if there is no communication with the Proxy servers.
- Call Restriction – (when Proxy isn't used), reject all outgoing Tel→IP calls that are associated with the destination IP address: 0.0.0.0.
- IP Security – When the IP Security feature is enabled (SecureCallFromIP = 1), the VoIP gateway accepts only those IP→Tel calls with a source IP address identical to one of the IP addresses entered in the Tel to IP Routing Table.
- Filter Calls to IP – When a Proxy is used, the gateway checks the Tel→IP routing table before a telephone number is routed to the Proxy. If the number is not allowed (number isn't listed or a Call Restriction routing rule was applied), the call is released.
- Always Use Routing Table – When this feature is enabled (AlwaysUseRouteTable = 1), even if a Proxy server is used, the SIP URI host name in the sent INVITE message is obtained from this table. Using this feature, users are able to assign a different SIP URI host name for different called and/or calling numbers.
- Assign Profiles to destination address (also when a Proxy is used).
- Alternative Routing – (When Proxy isn't used) an alternative IP destination for telephone number prefixes is available. To associate an alternative IP address to called telephone number prefix, assign it with an additional entry (with a different IP address), or use an FQDN that resolves to two IP addresses. Call is sent to the alternative destination when one of the following occurs:
    - No ping to the initial destination is available, or when poor QoS (delay or packet loss, calculated according to previous calls) is detected, or when a DNS host name is not resolved. For detailed information on Alternative Routing, refer to Section 9.8 on page 284.
    - When a release reason that is defined in the 'Reasons for Alternative Tel to IP Routing' table is received. For detailed information on the 'Reasons for Alternative Routing Tables', refer to Section 5.5.4.6 on page 113.

  Alternative routing (using this table) is commonly implemented when there is no response to an INVITE message (after INVITE retransmissions). The gateway then issues an internal 408 'No Response' implicit release reason. If this reason is included in the 'Reasons for Alternative Routing' table, the gateway immediately initiates a call to the redundant destination using the next matched entry in the 'Tel to IP Routing' table. Note that if a domain name in this table is resolved to two IP addresses, the timeout for INVITE retransmissions can be reduced by using the parameter 'Number of RTX Before Hotswap'.

Note: If the alternative routing destination is the gateway itself, the call can be configured to be routed back to PSTN. This feature is referred to as 'PSTN Fallback', meaning that if sufficient voice quality is not available over the IP network, the call is routed through legacy telephony system (PSTN).

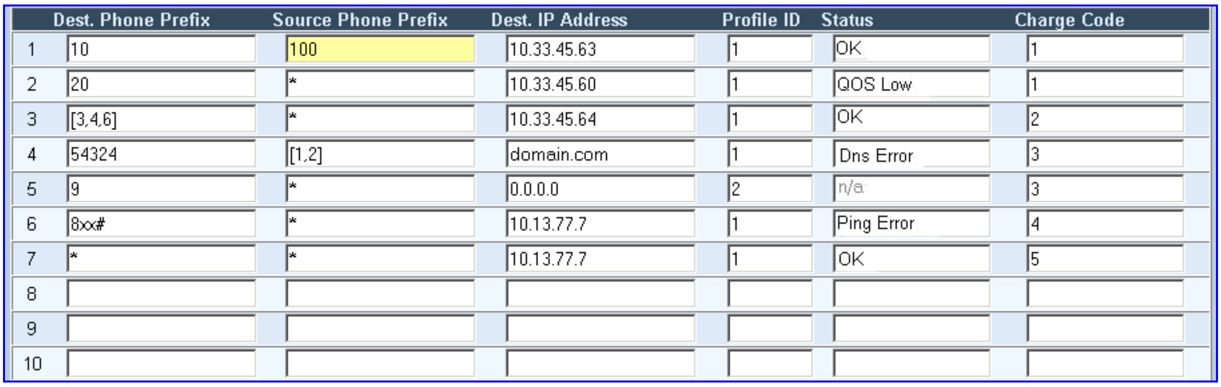
Tip: Tel to IP routing can be performed either before or after applying the number manipulation rules. To control when number manipulation is performed, set the Tel to IP Routing Mode (or RouteModeTel2IP *ini* file) parameter (described in Table 5-15).

You can also configure the Tel to IP Routing tbale using the *ini* file parameter Prefix (refer to Section 6.5.10 on page 242).

➢ To configure the Tel to IP Routing table, take these 6 steps:

1. Open the 'Tel to IP Routing' screen (**Protocol Management** menu > **Routing Tables** submenu > **Tel to IP Routing** option); the 'Tel to IP Routing' screen is displayed (shown in Figure 5-17).

Figure 5-17: Tel to IP Routing Table Screen



	Dest. Phone Prefix	Source Phone Prefix	Dest. IP Address	Profile ID	Status	Charge Code
1	10	100	10.33.45.63	1	OK	1
2	20	*	10.33.45.60	1	QOS Low	1
3	[3,4,6]	*	10.33.45.64	1	OK	2
4	54324	[1,2]	domain.com	1	Dns Error	3
5	9	*	0.0.0.0	2	n/a	3
6	8xx#	*	10.13.77.7	1	Ping Error	4
7	*	*	10.13.77.7	1	OK	5
8						
9						
10						

2. In the 'Tel to IP Routing Mode' field, select the Tel to IP routing mode (refer to Table 5-15).
3. In the 'Routing Index' drop-down list, select the range of entries that you want to edit.
4. Configure the Tel to IP Routing table according to Table 5-15.
5. Click the **Submit** button to save your changes.
6. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Table 5-15: Tel to IP Routing Table (continues on pages 108 to 109)

Parameter	Description
Tel to IP Routing Mode **[RouteModeTel2IP]**	Defines order between routing incoming calls to IP, using routing table, and manipulation of destination number. Route calls before manipulation **[0]** = Tel→IP calls are routed before the number manipulation rules are applied (default). Route calls after manipulation **[1]** = Tel→IP calls are routed after the number manipulation rules are applied. **Note:** Not applicable if Proxy routing is used.

Table 5-15: Tel to IP Routing Table (continues on pages 108 to 109)

Parameter	Description
Dest. Phone Prefix	Each entry in the Destination Phone Prefix fields represents a called telephone number prefix. The prefix can be 1 to 19 digits long. An asterisk (*) represents all numbers.
Source Phone Prefix	Each entry in the Source Phone Prefix fields represents a calling telephone number prefix. The prefix can be 1 to 19 digits long. An asterisk (*) represents all numbers.
Any telephone number whose destination number matches the prefix defined in the 'Destination Phone Prefix' field *and* its source number matches the prefix defined in the adjacent 'Source Phone Prefix' field, is sent to the IP address entered in the 'IP Address' field. Note that Tel to IP routing can be performed according to a combination of source and destination phone prefixes, or using each independently. **Note 1:** An additional entry of the same prefixes can be assigned to enable alternative routing. **Note 2:** For available notations that represent multiple numbers, refer to Section 5.5.3.1 on page 102.	
Dest. IP Address	In each of the IP Address fields, enter the IP address (and optionally port number) that is assigned to these prefixes. Domain names, such as domain.com, can be used instead of IP addresses. For example: <IP Address>:<Port> To discard outgoing IP calls, enter 0.0.0.0 in this field. **Note:** When using domain names, you must enter a DNS server IP address, or alternatively define these names in the 'Internal DNS Table'.
Profile ID	Enter the number of the IP profile that is assigned to the destination IP address defined in the 'Destination IP Address' field.
Status	A read only field representing the quality of service of the destination IP address. N/A = Alternative Routing feature is disabled. OK = IP route is available Ping Error = No ping to IP destination, route is not available QoS Low = Bad QoS of IP destination, route is not available DNS Error = No DNS resolution (only when domain name is used instead of an IP address).
Charge Code	An optional Charge Code (1 to 25) can be applied to each routing rule to associate it with an entry in the Charge Code table (refer to Section 5.5.2.3.1 on page 96).

5.5.4.3 IP to Trunk Group Routing

The IP to Trunk Group Routing Table is used to route incoming IP calls to groups of channels (for digital modules, these are E1/T1 B-channels) called trunk groups. Calls are assigned to trunk groups according to any combination of the following three options (or using each independently):

- Destination phone prefix
- Source phone prefix
- Source IP address

The call is then sent to the VoIP gateway channels assigned to that trunk group. The specific channel, within a trunk group that is assigned to accept the call is determined according to the trunk group's channel selection mode which is defined in the Trunk Group Settings screen (Section 5.5.10 on page 132) or according to the global parameter 'ChannelSelectMode' (refer to Table 5-7 on page 85). Trunk groups can be used on both FXO and FXS modules; however, usually they are used with FXO modules.

Note: When a release reason that is defined in the ‘Reasons for Alternative IP to Tel Routing’ table is received for a specific IP→Tel call, an alternative trunk group for that call is available. To associate an alternative trunk group to an incoming IP call, assign it with an additional entry in the ‘IP to Trunk Group Routing’ table (repeat the same routing rules with a different trunk group ID). For detailed information on the ‘Reasons for Alternative Routing Tables’, refer to Section 5.5.4.5 on page 112.

To use trunk groups you must also do the following.

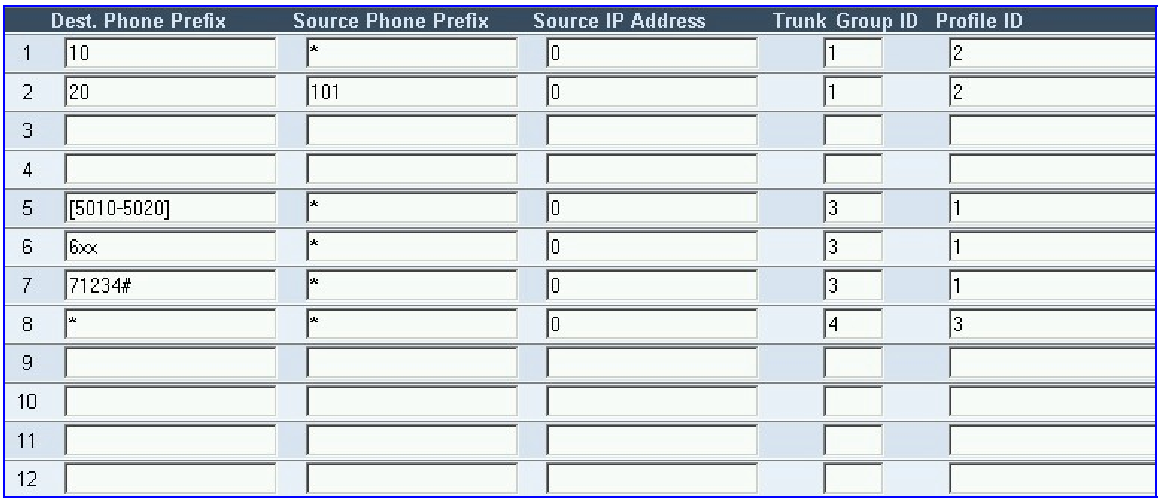
- You must assign a trunk group ID to the VoIP gateway channels in the Trunk Group Table screen. For information on how to assign a trunk group ID to a channel, refer to Section 5.5.9 on page 131.
- You can configure the Trunk Group Settings table to determine the method in which new calls are assigned to channels within the trunk groups (a different method for each trunk group can be configured). For information on how to enable this option, refer to Section 5.5.10 on page 132. If a Channel Select Mode for a specific trunk group isn’t specified, then the global ‘Channel Select Mode’ parameter (defined in ‘General Parameters’ screen under ‘Advanced Parameters’) applies.

You can also configure the IP to Trunk Group Routing table using the *ini* file parameter PSTNPrefix (refer to Section 6.5.10 on page 242).

➢ To configure the IP to Trunk Group Routing table, take these 6 steps:

1. Open the ‘IP to Trunk Group Routing’ screen (**Protocol Management** menu > **Routing Tables** submenu > **IP to Trunk Group Routing** option); the ‘IP to Trunk Group Routing’ table screen is displayed (shown in Figure 5-18).

Figure 5-18: IP to Trunk Group Routing Table Screen



	Dest. Phone Prefix	Source Phone Prefix	Source IP Address	Trunk Group ID	Profile ID
1	10	*	0	1	2
2	20	101	0	1	2
3					
4					
5	[5010-5020]	*	0	3	1
6	6xx	*	0	3	1
7	71234#	*	0	3	1
8	*	*	0	4	3
9					
10					
11					
12					

2. In the ‘IP to Tel Routing Mode’ field, select the IP to Tel routing mode (refer to Table 5-16).
3. In the ‘Routing Index’ drop-down list, select the range of entries that you want to edit (up to 24 entries can be configured).
4. Configure the IP to Trunk Group Routing table according to Table 5-16.
5. Click the **Submit** button to save your changes.
6. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Table 5-16: IP to Trunk Group Routing Table

Parameter	Description
IP to Tel Routing Mode **[RouteModeIP2Tel]**	Defines order between routing calls to Trunk group and manipulation of destination number. Route calls before manipulation **[0]** = IP→Tel calls are routed before the number manipulation rules are applied (default). Route calls after manipulation **[1]** = IP→Tel calls are routed after the number manipulation rules are applied.
Dest. Phone Prefix	Each entry in the Destination Phone Prefix fields represents a called telephone number prefix. The prefix can be 1 to 49 digits long. An asterisk (*) represents all numbers.
Source Phone Prefix	Each entry in the Source Phone Prefix fields represents a calling telephone number prefix. The prefix can be 1 to 49 digits long. An asterisk (*) represents all numbers.
Source IP Address	Each entry in the Source IP Address fields represents the source IP address of an IP→Tel call (obtained from the Contact header in the INVITE message). **Note:** The source IP address can include the 'x' wildcard to represent single digits. For example: 10.8.8.xx represents all the addresses between 10.8.8.10 to 10.8.8.99.
Any SIP incoming call whose destination number matches the prefix defined in the 'Destination Phone Prefix' field *and* its source number matches the prefix defined in the adjacent 'Source Phone Prefix' field *and* its source IP address matches the address defined in the 'Source IP Address' field, is assigned to the trunk group entered in the field to the right of these fields. Note that IP to hunt group routing can be performed according to any combination of source / destination phone prefixes and source IP address, or using each independently. **Note:** For available notations that represent multiple numbers (used in the prefix columns), refer to Section 5.5.3.1 on page 102.	
Trunk Group ID	In each of the Trunk Group ID fields, enter the trunk group ID to which calls that match these prefixes are assigned.
Profile ID	Enter the number of the IP profile that is assigned to the routing rule.

5.5.4.4 Internal DNS Table

The internal DNS table, similar to a DNS resolution, translates host names into IP addresses. This table is used when hostname translation is required (e.g., 'Tel to IP Routing' table). Two different IP addresses can be assigned to the same hostname. If the hostname isn't found in this table, the gateway communicates with an external DNS server.

Assigning two IP addresses to hostname can be used for alternative routing (using the 'Tel to IP Routing' table).

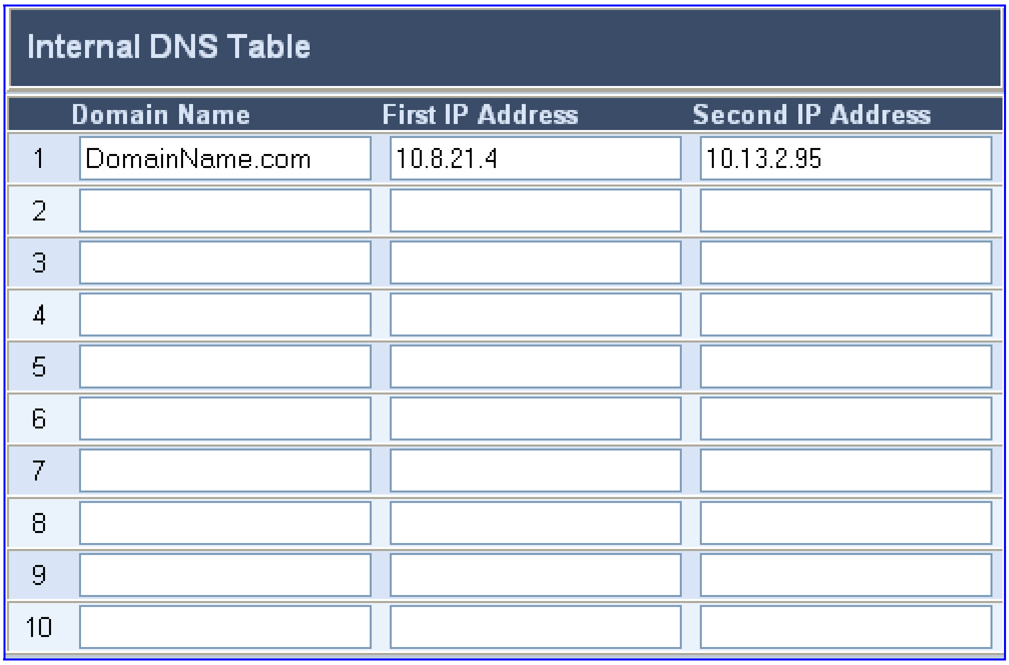
For a description of the *ini* file parameter (DNS2IP) used to configure the Internal DNS table, refer to Section 6.5.1 on page 215.

Note: If the Internal DNS table is configured, the gateway first tries to resolve a domain name using this table. If the domain name isn't found, the gateway performs a DNS resolution using an external DNS server.

➢ To configure the internal DNS table, take these 7 steps:

1. Open the ‘Internal DNS Table’ screen (**Protocol Management** menu > **Routing Tables** submenu > **Internal DNS Table** option); the ‘Internal DNS Table’ screen is displayed.

Figure 5-19: Internal DNS Table Screen



Internal DNS Table

	Domain Name	First IP Address	Second IP Address
1	DomainName.com	10.8.21.4	10.13.2.95
2			
3			
4			
5			
6			
7			
8			
9			
10			

2. In the ‘Domain Name' field, enter the host name to be translated. You can enter a string up to 31 characters long.
3. In the ‘First IP Address’ field, enter the first IP address (in dotted format notation) that the hostname is translated to.
4. In the ‘Second IP Address’ field, enter the second IP address that the hostname is translated to.
5. Repeat steps 2 to 4, for each Internal DNS Table entry.
6. Click the **Submit** button to save your changes.
7. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

5.5.4.5 Internal SRV Table

The Internal SRV table is used for resolving host names to DNS A-Records. Three different A-Records can be assigned to a hostname. Each A-Record contains the host name, priority, weight, and port.

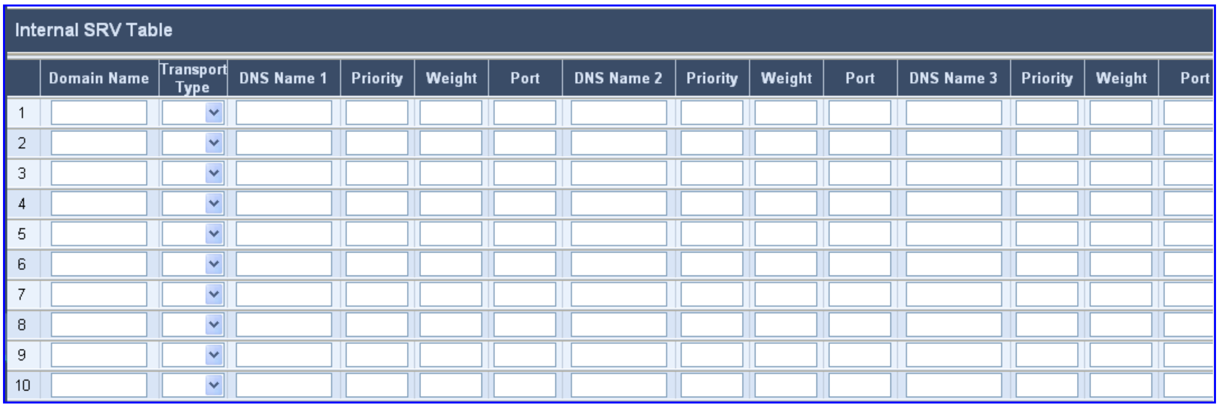
You can also configure the Internal SRV table using the *ini* file parameter SRV2IP (refer to Section 6.5.1 on page 215).

Note: If the Internal SRV table is configured, the gateway first tries to resolve a domain name using this table. If the domain name isn’t found, the gateway performs an SRV resolution using an external DNS server.

➢ To configure the Internal SRV table, take these 9 steps:

1. Open the 'Internal SRV Table' screen (**Protocol Management** menu > **Routing Tables** submenu > **Internal SRV Table** option); the 'Internal SRV Table' screen is displayed.

Figure 5-20: Internal SRV Table Screen



Internal SRV Table

	Domain Name	Transport Type	DNS Name 1	Priority	Weight	Port	DNS Name 2	Priority	Weight	Port	DNS Name 3	Priority	Weight	Port
1														
2														
3														
4														
5														
6														
7														
8														
9														
10														

2. In the 'Domain Name' field, enter the hostname to be translated. You can enter a string up to 31 characters long.
3. From the 'Transport Type' drop-down list, select a transport type.
4. In the 'DNS Name 1' field, enter the first DNS A-Record to which the hostname is translated.
5. In the 'Priority', 'Weight' and 'Port' fields, enter the relevant values
6. Repeat steps 4 to 5, for the second and third DNS names, if required.
7. Repeat steps 2 to 6, for each Internal SRV Table entry.
8. Click the **Submit** button to save your changes.
9. To save the changes so they are available after a hardware reset or power fail, refer to Section 5.9.2 on page 200.

5.5.4.6 Reasons for Alternative Routing

The Reasons for Alternative Routing screen includes two tables (Tel→IP and IP→Tel). Each table enables you to define up to four different release reasons. If a call is released as a result of one of these reasons, the gateway tries to find an alternative route to that call. The release reason for IP→Tel calls is provided in Q.931 notation. The release reason for Tel→IP calls is provided in SIP 4xx, 5xx and 6xx response codes. For Tel→IP calls an alternative IP address, for IP→Tel calls an alternative hunt (analog modules) or trunk (digital modules) group.

Refer to Section 5.5.4.2 on page 107 for information on defining an alternative IP address. Refer to the Section 5.5.4.3 on page 109 for information on defining an alternative trunk group.

You can use this table, for example:

- For Tel→IP calls, when there is no response to an INVITE message (after INVITE retransmissions), and the gateway then issues an internal 408 'No Response' implicit release reason.
- For IP→Tel calls, when the destination is busy, and release reason #17 is issued or for other call releases that issue the default release reason (#3). Refer to 'DefaultReleaseCause' in Table 5-7.

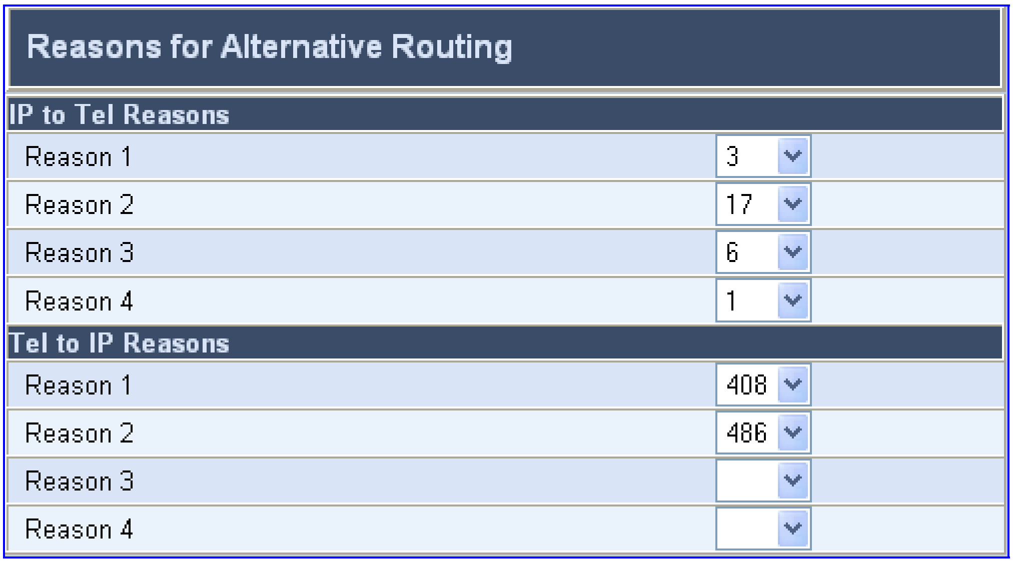
Note: The reasons for alternative routing option for Tel→IP calls only apply when Proxy isn't used.

For configuring this table, you can also use the *ini* file parameters AltRouteCauseTel2IP and AltRouteCauseIP2Tel (refer to Section 6.5.10 on page 242).

➢ To configure the reasons for alternative routing, take these 5 steps:

1. Open the 'Reasons for Alternative Routing' screen (**Protocol Management** menu > **Routing Tables** submenu > **Reasons for Alternative Routing** option); the 'Reasons for Alternative Routing' screen is displayed.

Figure 5-21: Reasons for Alternative Routing Screen



Reasons for Alternative Routing

IP to Tel Reasons	
Reason 1	3
Reason 2	17
Reason 3	6
Reason 4	1

Tel to IP Reasons	
Reason 1	408
Reason 2	486
Reason 3	
Reason 4	

2. In the 'IP to Tel Reasons' table, from the drop-down list select up to four different call failure reasons that invoke an alternative IP to Tel routing.
3. In the 'Tel to IP Reasons' table, from the drop-down list select up to four different call failure reasons that invoke an alternative Tel to IP routing.
4. Click the **Submit** button to save your changes.
5. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

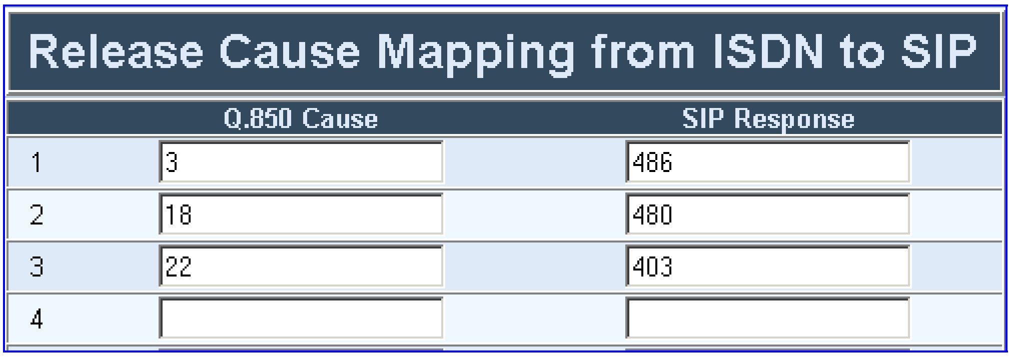
5.5.4.7 Release Cause Mapping

The Release Cause Mapping screen enables the Mediant 1000 to map (up to 12) different SIP Responses to Q.850 Release Causes and vice versa, thereby overriding the hard-coded mapping mechanism (described in Appendix H on page 445).

➢ To configure the release cause mapping, take these 5 steps:

1. Open the ‘Release Cause Mapping’ screen (**Protocol Management** menu > **Routing Tables** submenu > **Release Cause Mapping** option); the ‘Release Cause Mapping’ screen is displayed.

Figure 5-22: Release Cause Mapping from ISDN to SIP



Release Cause Mapping from ISDN to SIP

	Q.850 Cause	SIP Response
1	3	486
2	18	480
3	22	403
4		

2. In the ‘Release Cause Mapping from ISDN to SIP’ table, define (up to 12) different Q.850 Release Causes and the SIP Responses they are mapped to.
3. In the ‘Release Cause Mapping from SIP to ISDN’ table, define (up to 12) different SIP Responses and the Q.850 Release Causes they are mapped to.
4. Click the **Submit** button to save your changes.
5. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

5.5.5 Configuring the Profile Definitions

Utilizing the Profiles feature, the Mediant 1000 provides high-level adaptation when connected to a variety of equipment (from both Tel and IP sides) and protocols, each of which requires a different system behavior. Using Profiles, users can assign different Profiles (behavior) on a per-call basis, using the Tel to IP and IP to Trunk Group Routing tables, or associate different Profiles to the gateway’s endpoints (analog modules) or B-channels (digital modules). The Profiles contain parameters such as Coders, T.38 Relay, Voice and DTMF Gains, Silence Suppression, Echo Canceler, RTP DiffServ, Current Disconnect and more. The Profiles feature allows users to tune these parameters or turn them on or off, per source or destination routing and/or the specific gateway or its ports. For example, specific ports can be designated to have a profile which always uses G.711.

Each call can be associated with one or two Profiles: Tel Profile and (or) IP Profile. If both IP and Tel profiles apply to the same call, the coders and other common parameters of the preferred Profile (determined by the Preference option) are applied to that call. If the Preference of the Tel and IP Profiles is identical, the Tel Profile parameters are applied.

Use the Profile Definitions submenu to configure profiles:

- Coder Group Settings (refer to Section 5.5.5.1 on page 116)
- Tel Profile Settings (refer to Section 5.5.5.2 on page 117)
- IP Profile Settings (refer to Section 5.5.5.3 on page 119)

Note: The default values of the parameters in the Tel and IP Profiles are identical to the Web/*ini* file parameter values. If a value of a parameter is changed in the Web/*ini* file, it is automatically updated in the Profiles correspondingly. After any parameter in the Profile is modified by the user, modifications to parameters in the Web/*ini* file no longer impact that Profile.

5.5.5.1 Coder Group Settings

Use the Coder Group Settings screen to define up to four different coder groups. These coder groups are used in the Tel and IP Profile Settings screens to assign different coders to Profiles.

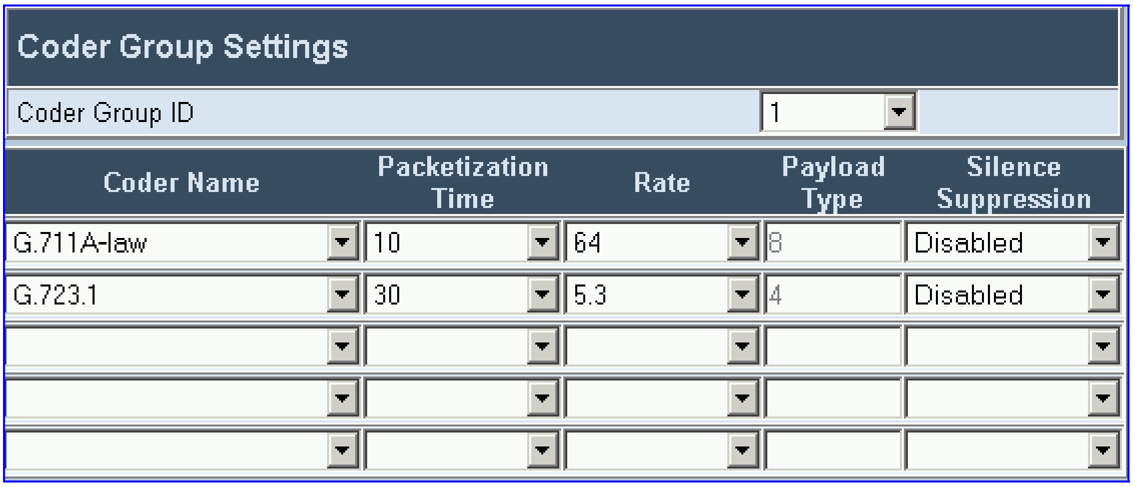
For each group you can define the first to fifth preferred coders (and their attributes) for the gateway. The first coder is the highest priority coder and is used by the gateway whenever possible. If the far end gateway cannot use the coder assigned as the first coder, the gateway attempts to use the next coder and so forth. For a list of coders supported by the gateway, refer to Table 5-5 on page 79.

You can also configure the coder groups using the *ini* file parameter CoderName_ID (refer to Section 6.5.7 on page 230).

➢ To configure the coder group settings, take these 11 steps:

1. Open the 'Coder Group Settings' screen (**Protocol Management** menu > **Profile Definitions** submenu > **Coder Group Settings** option); the 'Coder Group Settings' screen is displayed.

Figure 5-23: Coder Group Settings Screen



Coder Group Settings

Coder Group ID	1

Coder Name	Packetization Time	Rate	Payload Type	Silence Suppression
G.711A-law	10	64	8	Disabled
G.723.1	30	5.3	4	Disabled

2. From the Coder Group ID drop-down list, select the coder group you want to edit (up to four coder groups can be configured).
3. From the Coder Name drop-down list, select the coder you want to use. For the full list of available coders and their corresponding attributes, refer to Table 5-5 on page 79.
4. From the Packetization Time drop-down list, select the packetization time (in msec) for the coder you selected. The packetization time determines how many coder payloads are combined into a single RTP packet.
5. From the Rate drop-down list, select the bit rate (in kbps) for the coder you selected.
6. In the Payload Type field, if the payload type for the coder you selected is dynamic, enter a value from 0 to 120 (payload types of 'well-known' coders cannot be modified). The payload type identifies the format of the RTP payload.
7. From the Silence Suppression drop-down list, enable or disable the silence suppression option for the coder you selected.
8. Repeat steps 3 to 7 for the second to fifth coders (optional).
9. Repeat steps 2 to 8 for the second to fourth coder groups (optional).
10. Click the **Submit** button to save your changes.
11. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Notes:

- Each coder can appear only once.
- The ptime specifies the packetization time the gateway expects to receive. The gateway always uses the ptime requested by the remote side for sending RTP packets.
- Only the ptime of the first coder in the defined coder list is declared in INVITE / 200 OK SDP, even if multiple coders are defined.
- If not specified, the packetization time (ptime) gets a default value.
- For G.729 it is also possible to select silence suppression without adaptations.
- If the coder G.729 is selected and silence suppression is enabled (for this coder), the gateway includes the string 'annexb=no' in the SDP of the relevant SIP messages. If silence suppression is set to 'Enable w/o Adaptations', 'annexb=yes' is included. An exception to this logic is when the remote gateway is a Cisco device (IsCiscoSCEMode).

5.5.5.2 Tel Profile Settings

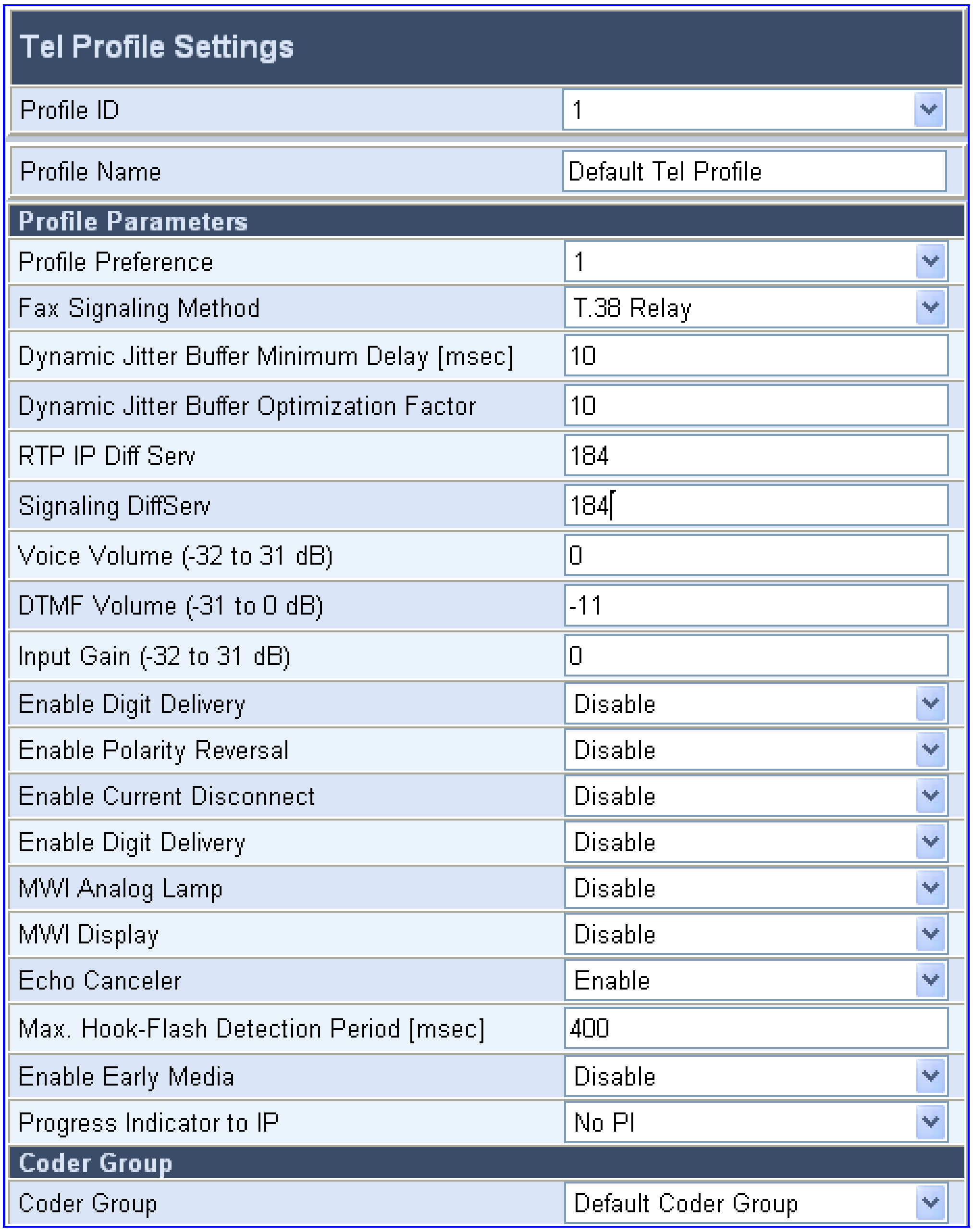
Use the Tel Profile Settings screen to define up to four different Tel Profiles. These Profiles are used in the 'Trunk Group' table to associate different Profiles to the gateway's channels, thereby applying different behavior to different channels.

You can also configure Tel Profiles using the *ini* file parameter TelProfile_ID (refer to Section 6.5.7 on page 230).

➢ To configure the Tel Profile settings, take these 9 steps:

1. Open the 'Tel Profile Settings' screen (**Protocol Management** menu > **Profile Definitions** submenu > **Tel Profile Settings** option); the 'Tel Profile Settings' screen is displayed.

Figure 5-24: Tel Profile Settings Screen



Tel Profile Settings

Profile ID	1
Profile Name	Default Tel Profile
Profile Parameters	
Profile Preference	1
Fax Signaling Method	T.38 Relay
Dynamic Jitter Buffer Minimum Delay [msec]	10
Dynamic Jitter Buffer Optimization Factor	10
RTP IP Diff Serv	184
Signaling DiffServ	184
Voice Volume (-32 to 31 dB)	0
DTMF Volume (-31 to 0 dB)	-11
Input Gain (-32 to 31 dB)	0
Enable Digit Delivery	Disable
Enable Polarity Reversal	Disable
Enable Current Disconnect	Disable
Enable Digit Delivery	Disable
MWI Analog Lamp	Disable
MWI Display	Disable
Echo Canceler	Enable
Max. Hook-Flash Detection Period [msec]	400
Enable Early Media	Disable
Progress Indicator to IP	No PI
Coder Group	
Coder Group	Default Coder Group

2. In the 'Profile ID' drop-down list, select the Tel Profile you want to edit (up to four Tel Profiles can be configured).

3. In the 'Profile Name' field, enter a name that enables you to identify the Profile intuitively and easily.

4. In the 'Profile Preference' drop-down list, select the preference (1-20) of the current Profile. The preference option is used to determine the priority of the Profile. Where '20' is the highest preference value. If both IP and Tel profiles apply to the same call, the coders and other common parameters (noted by an asterisk in the description of the parameter TelProfile_ID) of the preferred Profile are applied to that call. If the Preference of the Tel and IP Profiles is identical, the Tel Profile parameters are applied.
   Note: If the coder lists of both IP and Tel Profiles apply to the same call, an intersection of the coders is performed (i.e., only common coders remain). The order of the coders is determined by the preference.

5. Configure the Profile's parameters according to your requirements. For detailed information on each parameter, refer to the description of the screen in which it is configured as an individual parameter.

6. In the 'Coder Group' drop-down list, select the coder group you want to assign to that Profile. You can select the gateway's default coders (refer to Section 5.5.1.3 on page 78) or one of the coder groups you defined in the Coder Group Settings screen (refer to Section 5.5.5.1 on page 116).

7. Repeat steps 2 to 6 for the second to fifth Tel Profiles (optional).

8. Click the **Submit** button to save your changes.

9. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

5.5.5.3 IP Profile Settings

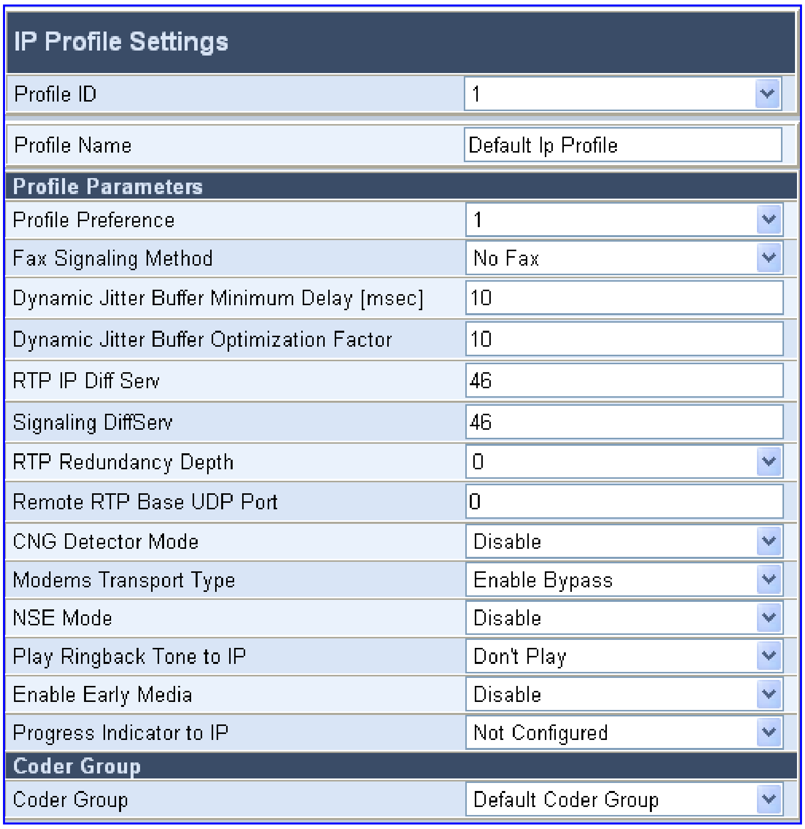
Use the IP Profile Settings screen to define up to four different IP Profiles. These Profiles are used in the Tel to IP and IP to Trunk Group Routing tables to associate different Profiles to routing rules. IP Profiles can also be used when working with Proxy server (set 'AlwaysUseRouteTable' to 1).

You can also configure the IP Profiles using the ini file parameter IPProfile_ID (refer to Section 6.5.7 on page 230).

➢ To configure the IP Profile settings, take these 9 steps:

1. Open the 'IP Profile Settings' screen (**Protocol Management** menu > **Profile Definitions** submenu > **IP Profile Settings** option); the 'IP Profile Settings' screen is displayed.

Figure 5-25: IP Profile Settings Screen



IP Profile Settings

Profile ID	1
Profile Name	Default Ip Profile
Profile Parameters	
Profile Preference	1
Fax Signaling Method	No Fax
Dynamic Jitter Buffer Minimum Delay [msec]	10
Dynamic Jitter Buffer Optimization Factor	10
RTP IP Diff Serv	46
Signaling DiffServ	46
RTP Redundancy Depth	0
Remote RTP Base UDP Port	0
CNG Detector Mode	Disable
Modems Transport Type	Enable Bypass
NSE Mode	Disable
Play Ringback Tone to IP	Don't Play
Enable Early Media	Disable
Progress Indicator to IP	Not Configured
Coder Group	
Coder Group	Default Coder Group

2. From the 'Profile ID' drop-down list, select the IP Profile you want to edit (up to four IP Profiles can be configured).

3. In the ‘Profile Name’ field, enter a name that enables you to identify the Profile intuitively and easily.
4. From the ‘Profile Preference’ drop-down list, select the preference (1-20) of the current Profile. The preference option is used to determine the priority of the Profile. Where ‘20’ is the highest preference value. If both IP and Tel profiles apply to the same call, the coders and other common parameters (noted by an asterisk in the description of the parameter IPProfile_ID) of the preferred Profile are applied to that call. If the Preference of the Tel and IP Profiles is identical, the Tel Profile parameters are applied.
   Note: If the coder lists of both IP and Tel Profiles apply to the same call, an intersection of the coders is performed (i.e., only common coders remain). The order of the coders is determined by the preference.
5. Configure the Profile’s parameters according to your requirements. For detailed information on each parameter, refer to the description of the screen in which it is configured as an individual parameter.
6. From the ‘Coder Group’ drop-down list, select the coder group you want to assign to that Profile. You can select the gateway’s default coders (refer to Section 5.5.1.3 on page 78) or one of the coder groups you defined in the Coder Group Settings screen (refer to Section 5.5.5.1 on page 116).
7. Repeat steps 2 to 6 for the second to fifth IP Profiles (optional).
8. Click the **Submit** button to save your changes.
9. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

5.5.6 Configuring the Endpoint Settings

The Endpoint Settings screens enable you to configure port-specific parameters.

Note: The Endpoint Settings screens are only applicable to the analog modules.

5.5.6.1 Authentication

The Authentication screen (normally used with FXS modules) defines a username and password combination for authentication for each Mediant 1000 port.

The ‘Authentication Mode’ parameter (described in Section 5.5.1.2 on page 70) determines if authentication is performed per port or for the entire gateway. If authentication is performed for the entire gateway, this table is ignored.

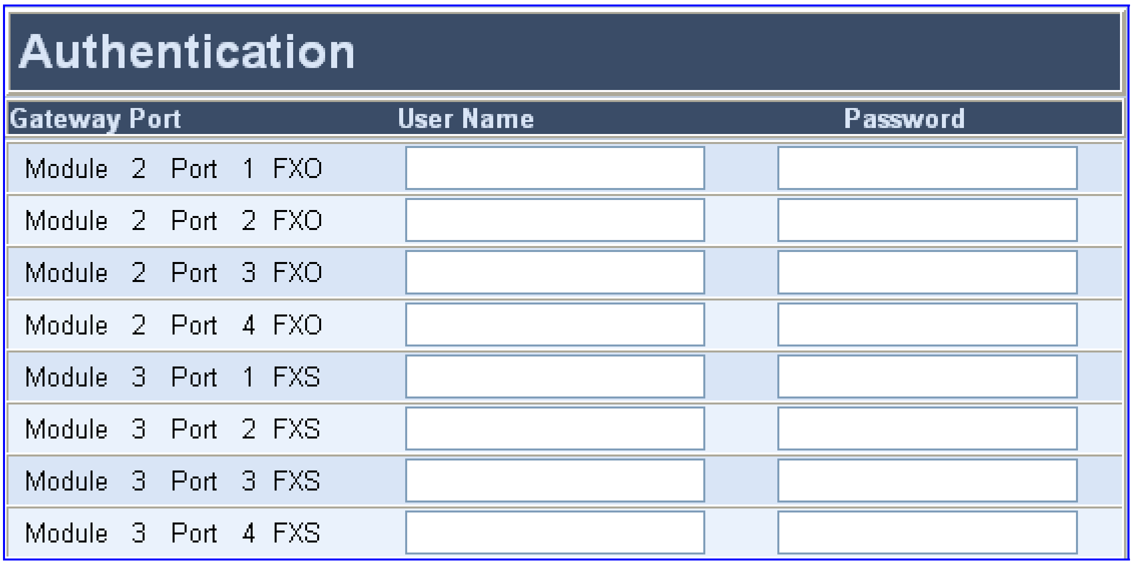
You can also configure the Authentication using the *ini* file parameter Authentication_x (refer to Section 6.5.7 on page 230).

Note: If either the username or password field is omitted, the port’s phone number (defined in Section 5.5.9 on page 131) and global password (refer to the parameter ‘Password’ described in Section 5.5.1.2 on page 70) are used instead.

➢ To configure the Authentication Table, take these 6 steps:

1. Set the 'Authentication Mode' parameter to 'Per Endpoint' (refer to Section 5.5.1.2 on page 70).
2. Open the 'Authentication' screen (**Protocol Management** menu > **Endpoint Settings** > **Authentication** option); the 'Authentication' screen is displayed.

Figure 5-26: Authentication Screen



Authentication

Gateway Port	User Name	Password
Module 2 Port 1 FXO		
Module 2 Port 2 FXO		
Module 2 Port 3 FXO		
Module 2 Port 4 FXO		
Module 3 Port 1 FXS		
Module 3 Port 2 FXS		
Module 3 Port 3 FXS		
Module 3 Port 4 FXS		

3. In the 'User Name' and 'Password' fields for a port, enter the username and password combination respectively.
4. Repeat Step 3 for each port.
5. Click the **Submit** button to save your changes.
6. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

5.5.6.2 Automatic Dialing

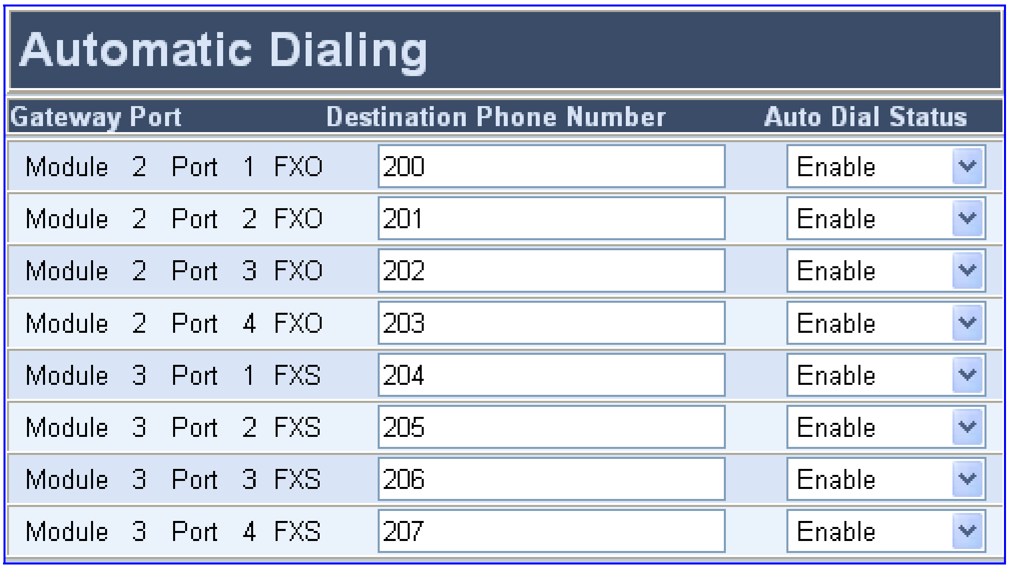
Use the Automatic Dialing Table to define telephone numbers that are automatically dialed when a specific port is used.

You can also configure the Automatic Dialing table using the *ini* file parameter TargetOfChannelX (refer to Section 6.5.10 on page 242).

➢ To configure the Automatic Dialing table, take these 6 steps:

1. Open the 'Automatic Dialing' screen (**Protocol Management** menu > **Endpoint Settings** submenu > **Automatic Dialing** option); the 'Automatic Dialing' screen is displayed.

Figure 5-27: Automatic Dialing Screen



Automatic Dialing

Gateway Port	Destination Phone Number	Auto Dial Status
Module 2 Port 1 FXO	200	Enable
Module 2 Port 2 FXO	201	Enable
Module 2 Port 3 FXO	202	Enable
Module 2 Port 4 FXO	203	Enable
Module 3 Port 1 FXS	204	Enable
Module 3 Port 2 FXS	205	Enable
Module 3 Port 3 FXS	206	Enable
Module 3 Port 4 FXS	207	Enable

2. In the 'Destination Phone Number' field for a port, enter the telephone number to dial.
3. In the 'Auto Dial Status' field, select one of the following:
   - Enable **[1]**: When a port is selected, when making a call, the number in the Destination Phone Number field is automatically dialed if phone is offhooked (for FXS modules) or ring signal is applied to port (FXO modules).
   - Disable **[0]**: The automatic dialing option on the specific port is disabled (the number in the Destination Phone Number field is ignored).
   - Hotline **[2]**: When a phone is offhooked and no digit is pressed for 'HotLineToneDuration', the number in the Destination Phone Number field is automatically dialed (applies to FXS and FXO modules).
4. Repeat step 3 for each port you want to use for Automatic Dialing.
5. Click the **Submit** button to save your changes.
6. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Notes:

- After a ring signal is detected, on an 'Enabled' FXO port, the gateway initiates a call to the destination number without seizing the line. The line is seized only after the call is answered.
- After a ring signal is detected on a 'Disabled' or 'Hotline' FXO port, the gateway seizes the line.

5.5.6.3 Caller ID

Use the Caller Display Information screen to send (to IP) Caller ID information when a call is made using the VoIP gateway (relevant to both FXS and FXO). The person receiving the call can use this information for caller identification. The information on this table is sent in an INVITE message in the 'From' header. For information on Caller ID restriction according to destination / source prefixes, refer to Section 5.5.3 on page 98.

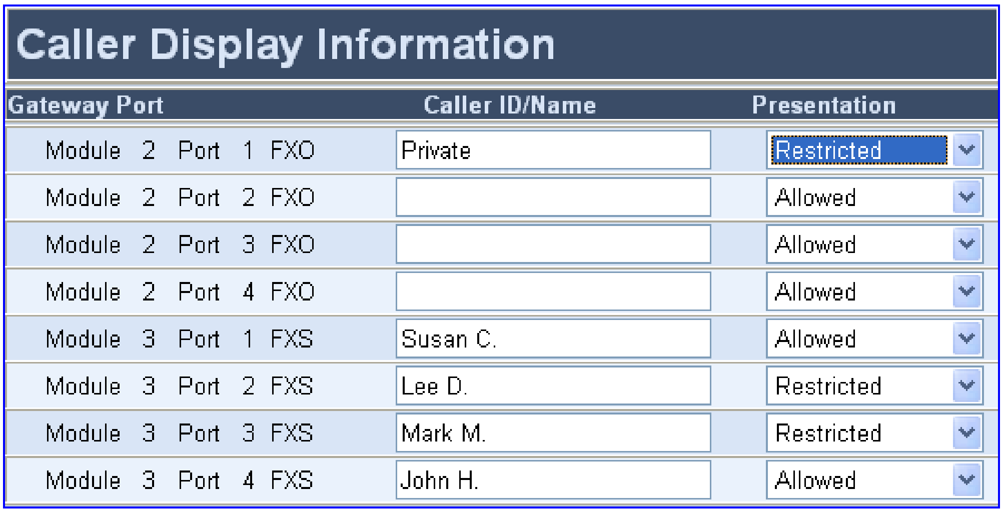
You can also configure the Caller Display Information table using the *ini* file parameter CallerDisplayInfoX (refer to Section 6.5.10 on page 242).

Note: If Caller ID name is detected on an FXO line (EnableCallerID = 1), it is used instead of the Caller ID name defined in this table (FXO modules only).

➢ To configure the Caller Display Information table, take these 6 steps:

1. Open the 'Caller Display Information' screen (**Protocol Management** menu > **Endpoint Settings** submenu > **Caller ID** option); the 'Caller Display Information' screen is displayed.

Figure 5-28: Caller Display Information Screen



Caller Display Information

Gateway Port	Caller ID/Name	Presentation
Module 2 Port 1 FXO	Private	Restricted
Module 2 Port 2 FXO		Allowed
Module 2 Port 3 FXO		Allowed
Module 2 Port 4 FXO		Allowed
Module 3 Port 1 FXS	Susan C.	Allowed
Module 3 Port 2 FXS	Lee D.	Restricted
Module 3 Port 3 FXS	Mark M.	Restricted
Module 3 Port 4 FXS	John H.	Allowed

2. In the Caller ID/Name field, enter the Caller ID string. The Caller ID string can contain up to 18 characters.

   Note that when the FXS module receives 'Private' or 'Anonymous' strings in the From header, it doesn't send the calling name or number to the Caller ID display.

3. In the 'Presentation' field, select 'Allowed' **[0]** to send the string in the Caller ID/Name field when a (Tel→IP) call is made using this VoIP gateway port. Select 'Restricted' **[1]** if you don't want to send this string.

Notes:

- When the 'Presentation' field is set to 'Restricted', the caller identity is passed to the remote side using only the P-Asserted-Identity and P-Preferred-Identity headers (AssertedIdMode).
- The value of the 'Presentation' field can (optionally) be overridden by configuring the 'Presentation' parameter in the 'Source Number Manipulation' table.

4. Repeat steps 2 and 3 for each VoIP gateway port.
5. Click the **Submit** button to save your changes.
6. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

5.5.6.4 Call Forward

The VoIP gateway allows you to forward incoming IP→Tel calls (using 302 response) based on the VoIP gateway port to which the call is routed (applicable only to FXS modules).

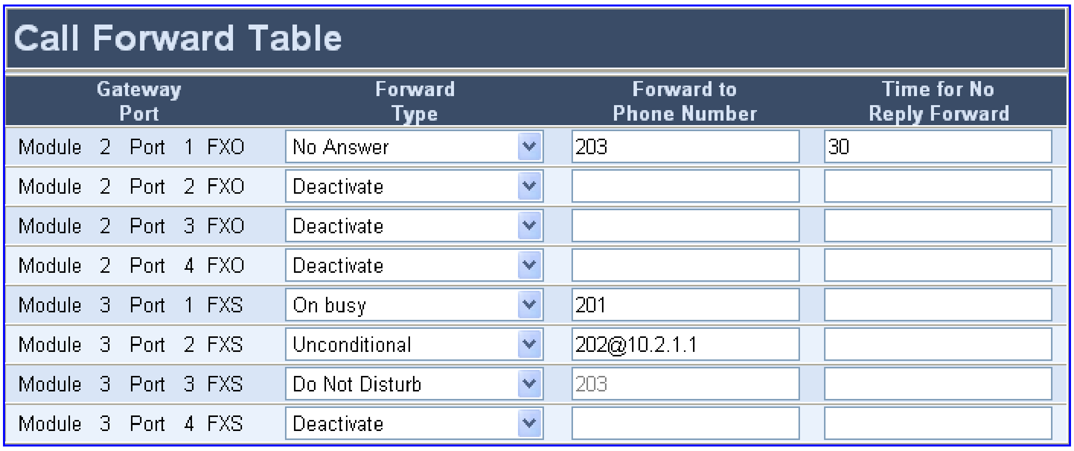
The Call Forwarding Table is applicable only if the Call Forward feature is enabled. To enable Call Forward set ‘Enable Call Forward’ to ‘Enable’ in the ‘Supplementary Services’ screen, or ‘EnableForward = 1’ in the *ini* file (refer to Table 5-8).

You can also configure the Call Forward table using the *ini* file parameter FwdInfo_x (refer to Section 6.5.10 on page 242).

➢ To configure the Call Forward table, take these 4 steps:

1. Open the ‘Call Forward Table’ screen (**Protocol Management** menu > **Endpoint Settings** submenu > **Call Forward** option); the ‘Call Forward Table’ screen is displayed.

Figure 5-29: Call Forward Table Screen



Call Forward Table

Gateway Port	Forward Type	Forward to Phone Number	Time for No Reply Forward
Module 2 Port 1 FXO	No Answer	203	30
Module 2 Port 2 FXO	Deactivate		
Module 2 Port 3 FXO	Deactivate		
Module 2 Port 4 FXO	Deactivate		
Module 3 Port 1 FXS	On busy	201	
Module 3 Port 2 FXS	Unconditional	202@10.2.1.1	
Module 3 Port 3 FXS	Do Not Disturb	203	
Module 3 Port 4 FXS	Deactivate		

2. Configure the Call Forward parameters for each port according to the table below.
3. Click the **Submit** button to save your changes.
4. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Table 5-17: Call Forward Table

Parameter	Description
Forward Type	Not in use **[0]** = Don't forward incoming calls (default). On Busy **[1]** = Forward incoming calls when the gateway port is busy. Immediate **[2]** = Forward any incoming call to the Phone number specified. No Answer **[3]** = Forward incoming calls that are not answered with the time specified in the 'Time for No Reply Forward' field. On busy or No reply **[4]** = Forward incoming calls when the port is busy or when calls are not answered after a configurable period of time. Do Not Disturb **[5]** = Immediately reject incoming calls.
Forward to Phone Number	Enter the telephone number or URI (number@IP address) to which the call is forwarded. **Note:** If this field only contains telephone number and Proxy isn't used, the 'forward to' phone number must be specified in the 'Tel to IP Routing' table of the forwarding gateway.
Time for No Reply Forward	If you have set the Forward Type for this port to No Answer, enter the number of seconds the VoIP gateway waits before forwarding the call to the phone number specified.

5.5.6.5 Caller ID Permissions

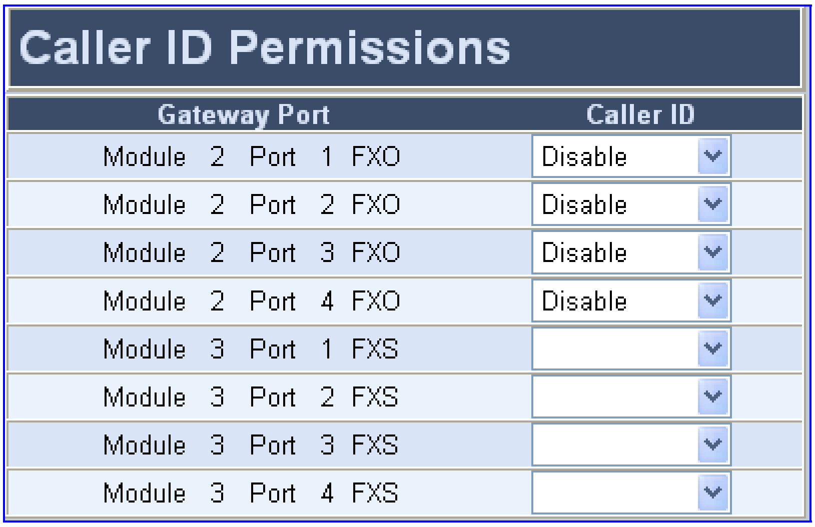
The Caller ID Permissions table is used to enable or disable (per port) the Caller ID generation (for FXS modules) and detection (for FXO modules). If a port isn't configured, its Caller ID generation / detection are determined according to the global parameter 'EnableCallerID' (described in Table 5-8).

You can also configure the Caller ID Permissions table using the *ini* file parameter EnableCallerID_X (refer to Section 6.5.10 on page 242).

➢ To configure the Caller ID Permissions Table, take these 5 steps:

1. Open the 'Caller ID Permissions' screen (**Protocol Management** menu > **Endpoint Settings** > **Caller ID Permissions** option); the 'Caller ID Permissions' screen is displayed.

Figure 5-30: Caller ID Permissions Screen



Caller ID Permissions

Gateway Port	Caller ID
Module 2 Port 1 FXO	Disable
Module 2 Port 2 FXO	Disable
Module 2 Port 3 FXO	Disable
Module 2 Port 4 FXO	Disable
Module 3 Port 1 FXS	
Module 3 Port 2 FXS	
Module 3 Port 3 FXS	
Module 3 Port 4 FXS	

2. In the ‘Caller ID’ field, select one of the following:
   - Enable: Enables Caller ID generation (FXS) or detection (FXO) for the specific port.
   - Disable: Caller ID generation (FXS) or detection (FXO) for the specific port is disabled.
   - Empty: Caller ID generation (FXS) or detection (FXO) for the specific port is determined according to the parameter ‘EnableCallerID’ (described in Table 5-8).
3. Repeat Step 2 for each port.
4. Click the **Submit** button to save your changes.
5. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

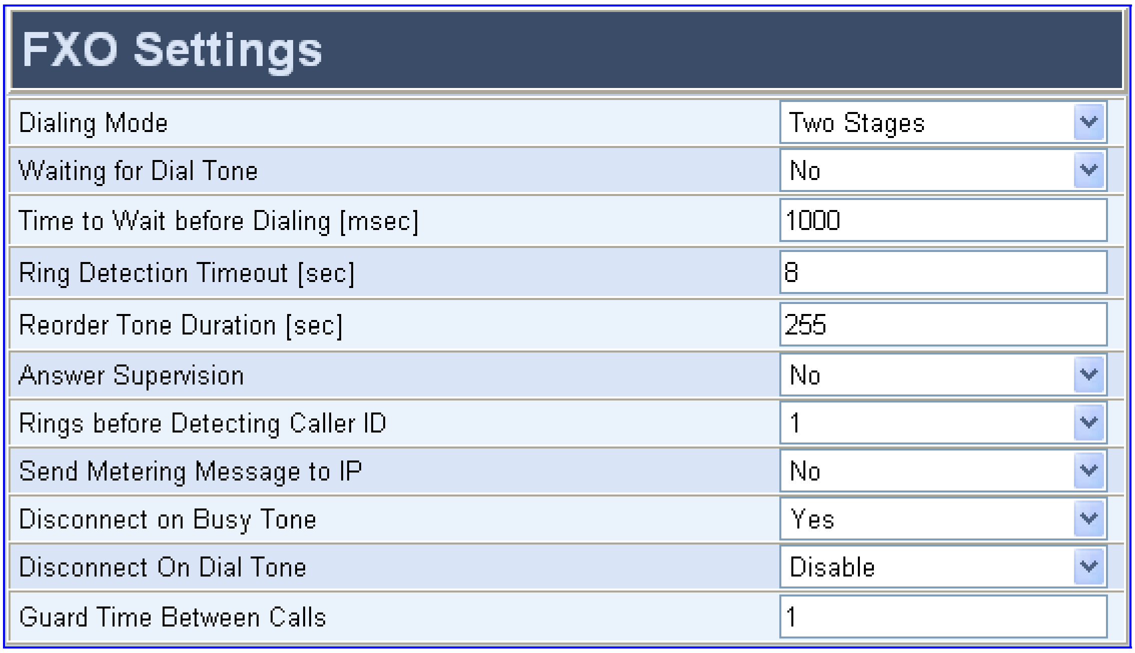
5.5.7 Configuring the FXO Parameters

Use this screen to configure the gateway’s specific FXO parameters.

➢ To configure the FXO parameters, take these 4 steps:

1. Open the ‘FXO Settings’ screen (**Protocol Management** menu > **FXO Settings** > **FXO Settings** option); the ‘FXO Settings’ screen is displayed.

Figure 5-31: FXO Settings Screen



FXO Settings

Dialing Mode	Two Stages
Waiting for Dial Tone	No
Time to Wait before Dialing [msec]	1000
Ring Detection Timeout [sec]	8
Reorder Tone Duration [sec]	255
Answer Supervision	No
Rings before Detecting Caller ID	1
Send Metering Message to IP	No
Disconnect on Busy Tone	Yes
Disconnect On Dial Tone	Disable
Guard Time Between Calls	1

2. Configure the FXO parameters according to Table 5-18.
3. Click the **Submit** button to save your changes.
4. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Table 5-18: FXO Parameters (continues on pages 127 to 129)

Parameter	Description
Dialing Mode **[IsTwoStageDial]**	One Stages **[0]** = One-stage dialing. Two Stages **[1]** = Two-stage dialing (default). Used for IP→FXO modules calls. If two-stage dialing is enabled, then the FXO module seizes one of the PSTN/PBX lines without performing any dial, the remote user is connected over IP to PSTN/PBX, and all further signaling (dialing and Call Progress Tones) is performed directly with the PBX without the gateway's intervention. If one-stage dialing is enabled, then the FXO module seizes one of the available lines (according to Channel Select Mode parameter), and dials the destination phone number received in INVITE message. Use the 'Waiting For Dial Tone' parameter to specify whether the dialing should come after detection of dial tone, or immediately after seizing of the line.
Waiting For Dial Tone **[IsWaitForDialTone]**	No **[0]** = Don't wait for dial tone. Yes **[1]** = Wait for dial tone (default). Used for IP→Mediant 1000/FXO modules, when 'One Stage Dialing' is enabled. If 'wait for dial tone' is enabled, the FXO module dials the phone number (to the PSTN/PBX line) only after it detects a dial tone. **Note 1:** The correct dial tone parameters should be configured in the Call Progress Tones file. **Note 2:** It can take the gateway 1 to 3 seconds to detect a dial tone (according to the dial tone configuration in the Call Progress Tones file). If 'Waiting For Dial Tone' is disabled, the FXO module immediately dials the phone number after seizing the PSTN/PBX line, without 'listening' to dial tone.
Time to Wait before Dialing [msec] **[WaitForDialTime]**	**Digital:** Determines the delay after hook-flash is generated and dialing is begun. Applies to call transfer (TrunkTransferMode = 3) on CAS gateways. **Analog:** Determines the delay before the gateway starts dialing on the FXO line in the following scenarios (applicable only to FXO modules): ▪ The delay between the time the line is seized and dialing is begun, during the establishment of an IP→Tel call. **Note:** Applicable only to FXO modules for single stage dialing, when waiting for dial tone (IsWaitForDialTone) is disabled. ▪ The delay between the time when Wink is detected and dialing is begun, during the establishment of an IP→Tel call (for DID lines, EnableDIDWink = 1). ▪ For call transfer. The delay after hook-flash is generated and dialing is begun. The valid range (in milliseconds) is 0 to 20,000 (i.e., 20 seconds). The default value is 1,000 (i.e., 1 second).
Ring Detection Timeout [sec] **[FXOBetweenRingTime]**	**Note:** Applicable only to FXO modules for Tel→IP calls. The Ring Detection timeout is used differently for normal and for automatic dialing. If automatic dialing is not used, and if Caller ID is enabled, then the FXO module seizes the line after detection of the second ring signal (allowing detection of caller ID, sent between the first and the second rings). If the second ring signal doesn't arrive for 'Ring Detection Timeout' the gateway doesn't initiate a call to IP. When automatic dialing is used, the FXO module initiates a call to IP when ringing signal is detected. The FXO line is seized only if the remote IP party answers the call. If the remote party doesn't answer the call and the ringing signal stops for 'Ring Detection Timeout', the FXO module Releases the IP call. Usually set to a value between 5 and 8. The default is 8 seconds.

Table 5-18: FXO Parameters (continues on pages 127 to 129)

Parameter	Description
Reorder Tone Duration [sec] **[TimeForReorderTone]**	▪ **Analog:** Busy or Reorder tone duration (seconds) the FXO module plays before releasing the line. The valid range is 0 to 100. The default is 10 seconds for digital gateways and 0 seconds for analog gateways. Usually, after playing a Reorder / Busy tone for the specified duration, the FXS module, starts playing an Offhook Warning tone. **Note 1:** Selection of Busy or Reorder tone is performed according to the release cause received from IP. **Note 2:** Refer also to the parameter 'CutThrough' (described in Table 5-7). ▪ **Digital:** Busy or Reorder Tone duration the CAS gateway plays before releasing the line. The valid range is 0 to 15. The default value is 10 seconds. Applicable also to ISDN if 'PlayBusyTone2ISDN = 2'. Selection of Busy or Reorder tone is done according to release cause received from IP.
Answer Supervision **[EnableVoiceDetection]**	Yes **[1]** = FXO/CAS module sends 200 OK (to INVITE) messages when speech/fax/modem is detected. No **[0]** = 200 OK is sent immediately after the FXO/CAS module finishes dialing (default). Usually this feature is used only when early media is used to establish voice path before the call is answered. **Note:** This feature is applicable only to 'One Stage' dialing for FXO modules.
Rings before Detecting Caller ID **[RingsBeforeCallerID]**	Sets the number of rings before the gateway starts detection of Caller ID (FXO only). 0 **[0]** = Before first ring. 1 **[1]** = After first ring (default). 2 **[2]** = After second ring.
Send Metering Message to IP **[SendMetering2IP]**	Enables or disables the sending of a metering tone INFO message to IP on detection of a metering pulse from the Tel side. This is applicable to FXO modules or digital modules when MFC/R2 is used. Valid options include: 0 = No (default). 1 = Yes. Analog interface: If SendMetering2IP is set to 1, the FXO modules send a metering tone INFO message to the IP upon detection of a 12/16 KHz metering pulse. FXS modules generate a 12/16 KHz metering tone upon reception of a metering message. Digital interface: If SendMetering2IP is set to 1, the digital modules send a metering tone INFO message to the IP upon detection of a MFCR/R2 metering pulse. **Note 1:** Suitable (12 or 16 KHz) coeff must be used for both FXS and FXO modules. The MeteringType parameter must be defined for both these modules. **Note 2:** The proprietary metering tone INFO message is shown in Section 13.1 on page 339.
Disconnect on Busy Tone **[DisconnectOnBusyTone]**	No **[0]** = Do not disconnect call on detection of busy tone (FXO/digital module). Yes **[1]** = Call is released (on FXO/digital modules) if busy or reorder (fast busy) tones are detected on the gateway's port (default).
Disconnect on Dial Tone **[DisconnectOnDialTone]**	FXO modules can disconnect a call after a dial tone from the PBX is detected. No **[0]** = Call isn't released. Yes **[1]** = Call is released if dial tone is detected on the gateway's FXO port (default). **Note:** This option is in addition to the mechanism that disconnects a call when either busy or reorder tones are detected.

Table 5-18: FXO Parameters (continues on pages 127 to 129)

Parameter	Description
Guard Time Between Calls **[GuardTimeBetweenCalls]**	Defines the time interval (in seconds) after a call has ended and a new call can be accepted for IP to Tel calls. Applicable only to FXO modules. The valid range is 0 to 10. The default value is 1 second. **Note:** Occasionally, after a call is ended and onhook is applied, a delay is required before placing a new call (and performing offhook). This is necessary to prevent wrong hook-flash detection or other glare phenomena.

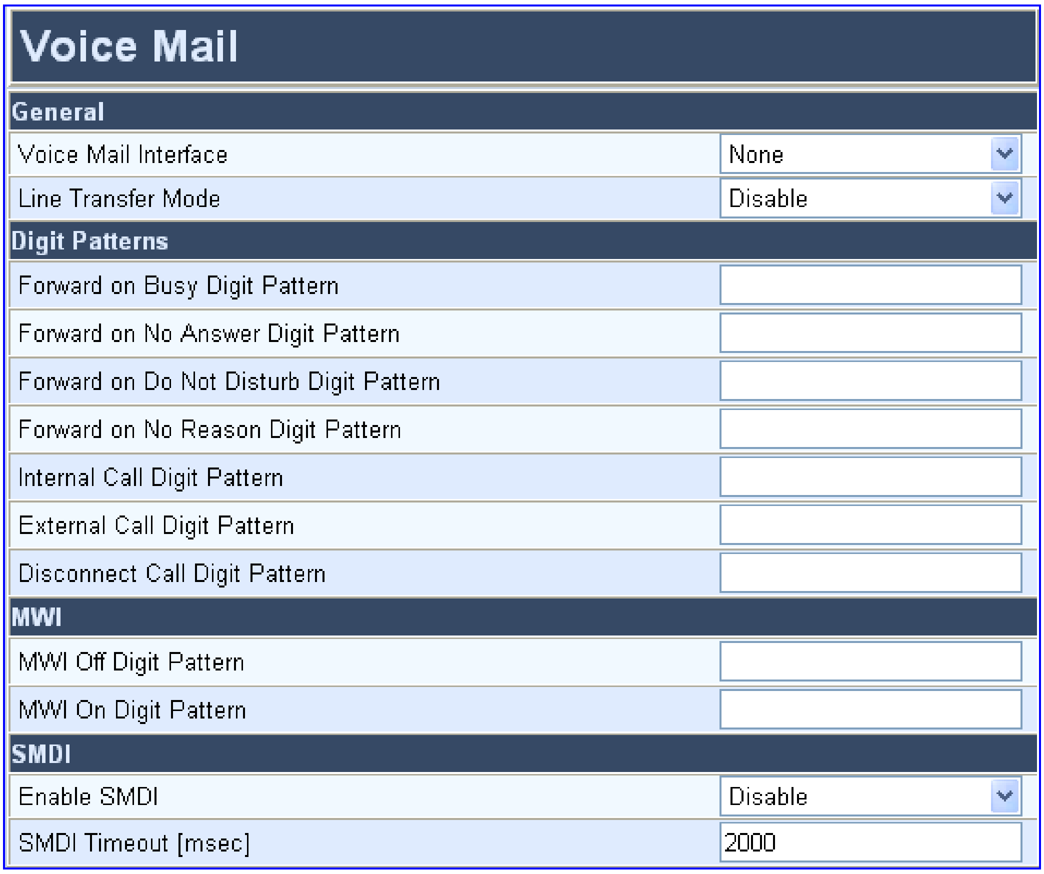
5.5.8 Configuring the Voice Mail (VM) Parameters

Use this screen to configure the Voice Mail (VM) parameters. The VM application applies only to FXO/CAS modules. For detailed information on VM, refer to the CPE Configuration Guide for Voice Mail.

➢ To configure the VM parameters, take these 4 steps:

1. Open the 'Voice Mail' screen (**Protocol Management** menu > **FXO Settings** > **Voice Mail** option); the 'Voice Mail' screen is displayed.

Figure 5-32: Voice Mail Screen



Voice Mail

General

Voice Mail Interface: None

Line Transfer Mode: Disable

Digit Patterns

Forward on Busy Digit Pattern:

Forward on No Answer Digit Pattern:

Forward on Do Not Disturb Digit Pattern:

Forward on No Reason Digit Pattern:

Internal Call Digit Pattern:

External Call Digit Pattern:

Disconnect Call Digit Pattern:

MWI

MWI Off Digit Pattern:

MWI On Digit Pattern:

SMDI

Enable SMDI: Disable

SMDI Timeout [msec]: 2000

2. Configure the Voice Mail parameters according to Table 5-19.
3. Click the **Submit** button to save your changes.
4. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Table 5-19: Voice Mail Parameters

Parameter	Description
General	
Voice Mail Interface **[VoiceMailInterface]**	Enables the VM application on the Mediant 1000 and determines the communication method used between the PBX and the gateway. None **[0]** (default) DTMF **[1]** SMDI **[2]** (N/A) QSIG **[3[** SETUP Only (ISDN) **[4]**
Line Transfer Mode **[LineTransferMode]**	Determines the transfer method used by the gateway. Disable **[0]** = IP (default). Blind Transfer **[1]** = PBX blind transfer. For analog interface, after receiving a REFER message from the IP side, the FXO sends a hook-flash to the PBX, dials the digits (that are received in the Refer-To header), and then immediately drops the line (on-hook). The PBX performs the transfer internally.
Digit Patterns The following digit pattern parameters apply only to VM applications that use the DTMF communication method. For the available patterns' syntaxes, refer to the CPE Configuration Guide for Voice Mail.	
Forward on Busy Digit Pattern **[DigitPatternForwardOnBusy]**	Determines the digit pattern used by the PBX to indicate 'call forward on busy'. The valid range is a 120-character string.
Forward on No Answer Digit Pattern **[DigitPatternForwardOnNoAnswer]**	Determines the digit pattern used by the PBX to indicate 'call forward on no answer'. The valid range is a 120-character string.
Forward on Do Not Disturb Digit Pattern **[DigitPatternForwardOnDND]**	Determines the digit pattern used by the PBX to indicate 'call forward on do not disturb'. The valid range is a 120-character string.
Forward on No Reason Digit Pattern **[DigitPatternForwardNoReason]**	Determines the digit pattern used by the PBX to indicate 'call forward with no reason'. The valid range is a 120-character string.
Internal Call Digit Pattern **[DigitPatternInternalCall]**	Determines the digit pattern used by the PBX to indicate an internal call. The valid range is a 120-character string.
External Call Digit Pattern **[DigitPatternExternalCall]**	Determines the digit pattern used by the PBX to indicate an external call. The valid range is a 120-character string.
Disconnect Call Digit Pattern **[TelDisconnectCode]**	Determines a digit pattern that, when received from the Tel side, indicates the gateway to disconnect the call. The valid range is a 25-character string.
MWI	
MWI Off Digit Pattern **[MWIOffCode]**	Determines a digit code used by the gateway to notify the PBX that there aren't any messages waiting for a specific extension. This code is added as prefix to the dialed number. The valid range is a 25-character string.
MWI On Digit Pattern **[MWIOnCode]**	Determines a digit code used by the gateway to notify the PBX of messages waiting for a specific extension. This code is added as prefix to the dialed number. The valid range is a 25-character string.
SMDI (currently not supported)	
Enable SMDI **[SMDI]**	N/A
SMDI Timeout **[SMDITimeOut]**	N/A

5.5.9 Configuring the Trunk Group Table

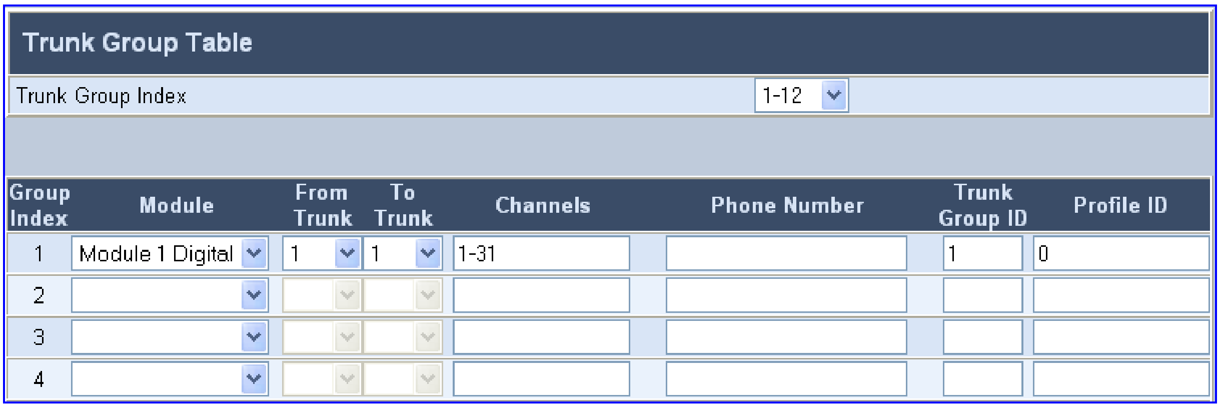
Use the Trunk Group table to assign trunk groups, profiles and logical telephone numbers to the gateway's channels. Trunk Groups are used for routing IP→Tel calls with common rules. Channels that are not defined are disabled.

You can also use the *ini* file parameter TrunkGroup_x to configure the Trunk Groups (refer to Section 6.5.10 on page 242).

➢ To configure the Trunk Group table, take these 4 steps:

1. Open the ‘Trunk Group Table’ screen (**Protocol Management** menu > **Trunk Group**); the ‘Trunk Group Table’ screen is displayed.

Figure 5-33: Trunk Group Table Screen



Trunk Group Table

Trunk Group Index 1-12

Group Index	Module	From Trunk	To Trunk	Channels	Phone Number	Trunk Group ID	Profile ID
1	Module 1 Digital	1	1	1-31		1	0
2							
3							
4							

2. Configure the Trunk Group according to Table 5-20.
3. Click the **Submit** button to save your changes.
4. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Table 5-20: Trunk Group Table (continues on pages 131 to 132)

Parameter	Description
Module	The module for which you want to define the trunk group. Valid options include: Module 1 Digital Module 2 FXO Module 3 FXS
From Trunk	Starting physical trunk number. **Note:** Applicable only to digital modules.
To Trunk	Ending physical trunk number. **Note:** Applicable only to digital modules.
Channels	The numbers in the Channels field represent the channels or ports on the back of the VoIP gateway (analog module), or B-channels (digital module). To enable the trunk’s channels, you **must** enter their number in this field. [n-m] represents a range of channels. For example, enter [1-24] to specify the channels from 1 to 24. **Note:** To represent all B-channels use a single asterisk instead.

Table 5-20: Trunk Group Table (continues on pages 131 to 132)

Parameter	Description
Phone Number	In each of the Phone Number fields, enter the first number in an ordered sequence that is assigned to the range of channels defined in the adjacent 'Channels' field. **Note:** This field is optional. The logical numbers defined in this field are used when an incoming PSTN / PBX call doesn't contain the calling number or called number (the latter being determined by the parameter 'ReplaceEmptyDstWithPortNumber'); these numbers are used to replace them. These logical numbers are also used for channel allocation for IP to Tel calls, if the trunk group's 'Channel Select Mode' is set to 'By Phone Number'.
Trunk Group ID	In each of the Trunk Group ID fields, enter the trunk group ID (1-99) assigned to the channels. The same trunk group ID can be used for more than one group of channels. Trunk group ID is used to define a group of common behavior channels that are used for routing IP to Tel calls. If an IP to Tel call is assigned to a trunk group, the call is routed to the channel or channels that correspond to the trunk group ID. You can configure the Trunk Group Settings table to determine the method in which new calls are assigned to channels within the trunk groups (refer to Section 5.5.10 on page 132). **Note:** You must configure the IP to Trunk Group Routing Table (assigns incoming IP calls to the appropriate trunk group). If you do not configure the IP to Trunk Group Routing Table, calls do not complete. For information on how to configure this table, refer to Section 5.5.4.3 on page 109.
Profile ID	Enter the number of the Tel profile that is assigned to the B-channels defined in the 'Channels' field.

5.5.10 Configuring the Trunk Group Settings

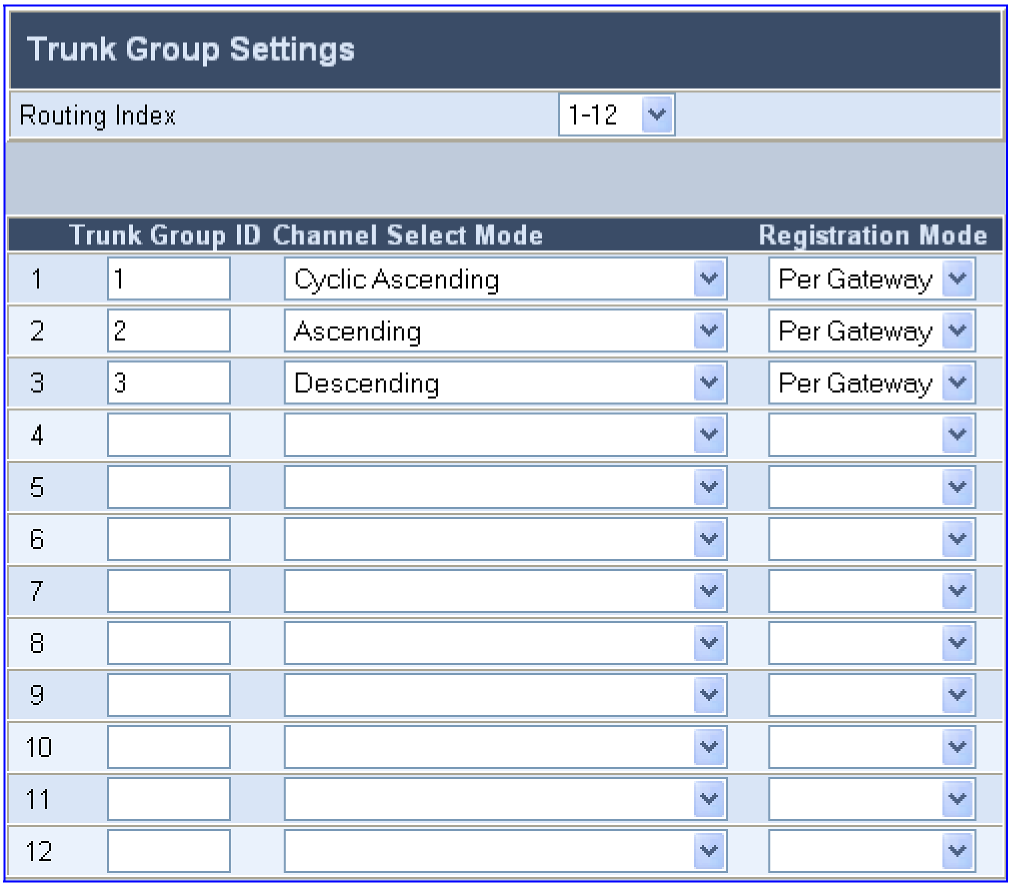
The Trunk Group Settings Table is used to determine the method in which new calls are assigned to channels within each trunk group. If such a rule doesn't exist (for a specific Trunk group), the global rule, defined by the Channel Select Mode parameter (**Protocol Definition** > **General Parameters**), applies.

You can also configure the Trunk Group Settings table using the *ini* file parameter TrunkGroupSettings (refer to Section 6.5.10 on page 242).

➢ To configure the Trunk Group Settings table, take these 7 steps:

1. Open the 'Trunk Group Settings' screen (**Protocol Management** menu > **Trunk Group Settings**); the 'Trunk Group Settings' screen is displayed.

Figure 5-34: Trunk Group Settings Screen



Trunk Group Settings

Routing Index: 1-12

	Trunk Group ID	Channel Select Mode	Registration Mode
1	1	Cyclic Ascending	Per Gateway
2	2	Ascending	Per Gateway
3	3	Descending	Per Gateway
4			
5			
6			
7			
8			
9			
10			
11			
12			

2. From the 'Routing Index' drop-down list, select the range of entries that you want to edit (up to 24 entries can be configured).
3. In the 'Trunk Group ID' field, enter the Trunk Group ID number.
4. From the 'Channel Select Mode' drop-down list, select the Channel Select Mode that determines the method in which new calls are assigned to channels within the Trunk groups entered in the field to the right of this field. For information on available Channel Select Modes, refer to Table 5-21.
5. Repeat steps 3 and 4, for each defined Trunk group.
6. Click the **Submit** button to save your changes.
7. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Table 5-21: Channel Select Modes

Mode	Description
By Dest Phone Number	Select the gateway port according to the called number (refer to the note below).
Cyclic Ascending	Select the next available channel in an ascending cycle order. Always select the next higher channel number in the Trunk Group. When the gateway reaches the highest channel number in the Trunk Group, it selects the lowest channel number in the Trunk Group and then starts ascending again (default).
Ascending	Select the lowest available channel. Always start at the lowest channel number in the Trunk Group and if that channel is not available, select the next higher channel.
Cyclic Descending	Select the next available channel in descending cycle order. Always select the next lower channel number in the Trunk Group. When the gateway reaches the lowest channel number in the Trunk Group, it selects the highest channel number in the Trunk Group and then start descending again.
Descending	Select the highest available channel. Always start at the highest channel number in the Trunk Group and if that channel is not available, select the next lower channel.
Dest Number + Cyclic Ascending	First select the gateway port according to the called number (refer to the note below). If the called number isn't found, then select the next available channel in ascending cyclic order. Note that if the called number is found, but the port associated with this number is busy, the call is released.
By Source Phone Number	Select the gateway port according to the calling number.

Note: The internal numbers of the gateway's channels are defined in the 'Trunk Group Table' under the 'Phone Number' column. For detailed information on the 'Trunk Group Table', refer to Section 5.5.9 on page 131).

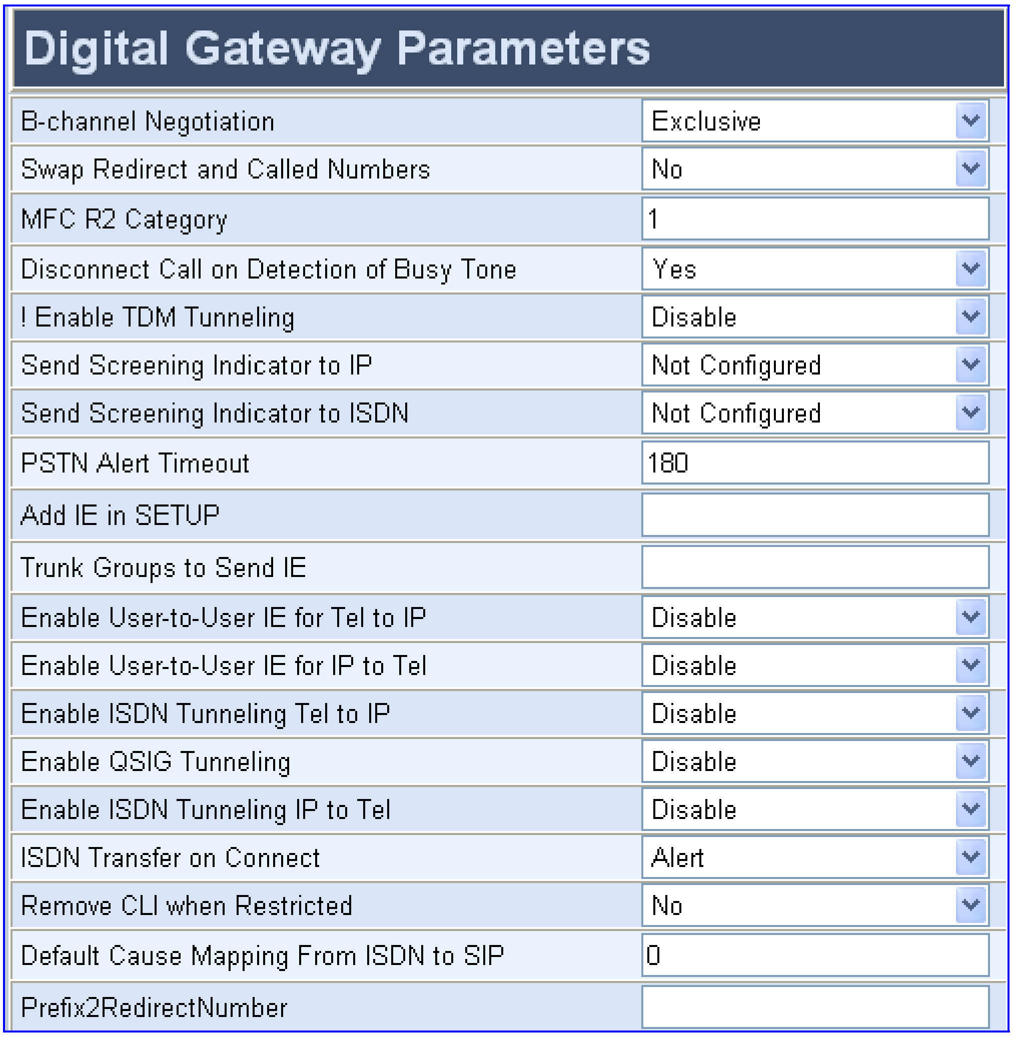
5.5.11 Configuring the Digital Gateway Parameters

The 'Digital Gateway' screen is used to configure miscellaneous digital parameters.

➢ To configure the digital gateway parameters, take these 4 steps:

1. Open the 'Digital Gateway Parameters' screen (**Protocol Management** menu > **Digital Gateway Parameters**); the 'Digital Gateway Parameters' screen is displayed.

Figure 5-35: Digital Gateway Parameters Screen



Digital Gateway Parameters

B-channel Negotiation	Exclusive
Swap Redirect and Called Numbers	No
MFC R2 Category	1
Disconnect Call on Detection of Busy Tone	Yes
! Enable TDM Tunneling	Disable
Send Screening Indicator to IP	Not Configured
Send Screening Indicator to ISDN	Not Configured
PSTN Alert Timeout	180
Add IE in SETUP	
Trunk Groups to Send IE	
Enable User-to-User IE for Tel to IP	Disable
Enable User-to-User IE for IP to Tel	Disable
Enable ISDN Tunneling Tel to IP	Disable
Enable QSIG Tunneling	Disable
Enable ISDN Tunneling IP to Tel	Disable
ISDN Transfer on Connect	Alert
Remove CLI when Restricted	No
Default Cause Mapping From ISDN to SIP	0
Prefix2RedirectNumber	

2. Configure the Digital Gateway parameters according to Table 5-22.
3. Click the **Submit** button to save your changes.
4. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Table 5-22: Digital Gateway Parameters (continues on pages 136 to 138)

Parameter	Description
B-channel Negotiation **[BchannelNegotiation]**	Determines the ISDN B-Channel negotiation mode. 0 = Preferred 1 = Exclusive (default) Applicable to ISDN protocols.
Swap Redirect and Called Numbers **[SwapRedirectNumber]**	0 = Don't change numbers (default) 1 = Incoming ISDN call that includes redirect number (sometimes referred as ‘original called number’) uses this number instead of the called number.
MFC R2 Category [**R2Category]**	MFC R2 Calling Party Category (CPC). The parameter provides information on calling party such as National or International call, Operator or Subscriber and Subscriber priority. The parameter range is 1 to 15, defining one of the MFC R2 tones.
Disconnect Call on Detection of Busy Tone **[DisconnectOnBusyTone]**	0 = Do not disconnect call on detection of busy tone 1 = Disconnect call on detection of busy tone (default). This parameter is applicable to CAS & ISDN protocols.
Enable TDM Tunneling **[EnableTDMoverIP]**	0 = Disabled (default). 1 = TDM Tunneling is enabled. When TDM Tunneling is enabled, the originating Mediant 1000 automatically initiates SIP calls from all enabled B-channels belonging to the E1/T1/J1 spans that are configured with the ‘Transparent’ protocol. The called number of each call is the internal phone number of the B-channel that the call originates from. The IP to Trunk Group routing table is used to define the destination IP address of the terminating Mediant 1000. The terminating Mediant 1000 gateway automatically answers these calls if its E1/T1 protocol is set to ‘Transparent’ (ProtocolType = 5).
Send Screening Indicator to IP **[ScreeningInd2IP]**	The parameter can overwrite the calling number screening indication for ISDN Tel→IP calls. -1 = not configured (interworking from ISDN to IP) or set to 0 for CAS. 0 = user provided, not screened. 1 = user provided, verified and passed. 2 = user provided, verified and failed. 3 = network provided. **Note:** Applicable only if Remote Party ID (RPID) header is enabled.
Send Screening Indicator to ISDN **[ScreeningInd2ISDN]**	Overwrites the screening indicator of the calling number for IP→Tel (ISDN) calls. -1 = Not Configured (interworking from IP to ISDN) (default). 0 = User provided, not screened. 1 = User provided, verified and passed. 2 = User provided, verified and failed. 3 = Network provided.
PSTN Alert Timeout **[PSTNAlertTimeout]**	Alert Timeout in seconds (ISDN T301 timer) for outgoing calls to PSTN. The default is 180 seconds. The range is 0 to 240. **Note:** The PSTN stack T301 timer can be overridden by a lower value, but it can’t be increased.
Add IE in SETUP **[AddIEinSetup]**	This parameter enables to add an optional Information Element data (in hex format) to ISDN SETUP message. For example: to add the following IE: ‘0x20,0x02,0x00,0xe1’, define: ‘AddIEinSetup = 200200e1’. **Note:** This IE is sent from the Trunk Group IDs defined by the parameter ‘SendIEonTG’.
Trunk Groups to Send IE **[SendIEonTG]**	A list of Trunk Group IDs (up to 50 characters) from where the optional ISDN IE, defined by the parameter ‘AddIEinSetup’, is sent. For example: ‘SendIEonTG = 1,2,4,10,12,6’.

Table 5-22: Digital Gateway Parameters (continues on pages 136 to 138)

Parameter	Description
Enable User-to-User IE for Tel to IP **[EnableUUITel2IP]**	0 = Disabled (default). 1 = Enable transfer of User-to-User Information Element (UUIE) from PRI Setup message to SIP INVITE message. The IE is transferred using a proprietary 'X-UserToUser' SIP header.
Enable User-to-User IE for IP to Tel **[EnableUUIIP2Tel]**	0 = Disabled (default). 1 = Enable transfer of (UUIE) from SIP INVITE message to PRI Setup message. The IE is received using a proprietary 'X-UserToUser' SIP header.
Enable ISDN Tunneling Tel to IP **[EnableISDNTunnelingTel2IP]**	Valid options include: 0 = Disable (default). 1 = Enable ISDN Tunneling from ISDN PRI to SIP using a proprietary SIP header 2 = Enable ISDN Tunneling from ISDN PRI to SIP using a dedicated message body When ISDN Tunneling is enabled, the gateway sends all ISDN PRI messages using the correlated SIP messages. Setup is tunneled using INVITE, all mid-call messages are tunneled using INFO, and Disconnect/Release is tunneled using BYE. The raw data from the ISDN is inserted into a proprietary SIP header (X-ISDNTunnelingInfo) or a dedicated message body (application/isdn) in the SIP messages. **Note:** It is necessary to set the parameter ISDNDuplicateQ931BuffMode to 128 (duplicate all messages) for this mechanism to function.
Enable QSIG Tunneling **[EnableQSIGTunneling]**	Enables QSIG tunneling over SIP according to <draft-elwell-sipping-qsig-tunnel-03>. 0 = Disable (default). 1 = Enable QSIG tunneling from QSIG to SIP and vice versa. When QSIG tunneling is enabled, all QSIG messages are sent as raw data in corresponding SIP messages using a dedicated message body. Note that QSIG tunneling must be enabled on both the originating and terminating gateways. **Note:** It is necessary to set the parameter 'ISDNDuplicateQ931BuffMode' to 128 (duplicate all messages) so that this mechanism can function.
Enable ISDN Tunneling IP to Tel **[EnableISDNTunnelingIP2Tel]**	Valid options include: 0 = Disable (default). 1 = Enable ISDN Tunneling from SIP to ISDN PRI using a proprietary SIP header. 2 = Enable ISDN Tunneling from SIP to ISDN PRI using a dedicated message body When ISDN Tunneling is enabled, the gateway extracts the raw data received in a proprietary SIP header (X-ISDNTunnelingInfo) or a dedicated message body (application/isdn) in the SIP messages and sends the data as ISDN messages to the PSTN side.
ISDN Transfer On Connect **[SendISDNTransferOnConnect]**	0 = Enable ISDN Transfer if outgoing call is in Alert state (default). 1 = Enable ISDN Transfer only if outgoing call is in Connect state. This parameter is used for the ECT/TBCT/RLT ISDN Transfer methods. Usually, the gateway requests the PBX to connect an incoming and an outgoing call. This parameter determines if the outgoing call (from the gateway to the PBX) must be connected before the transfer is initiated.
Remove CLI when Restricted **[RemoveCLIWhenRestricted]**	Determines (for IP to Tel calls) whether the Calling Number IE and Calling Name IE are removed from the outgoing ISDN Setup message if the presentation is set to Restricted. 0 = IE aren't removed (default). 1 = IE are removed.

Table 5-22: Digital Gateway Parameters (continues on pages 136 to 138)

Parameter	Description
Remove CLI when Restricted **[RemoveCLIWhenRestricted]**	Determines (for IP to Tel calls) whether the Calling Number IE and Calling Name IE are removed from the outgoing ISDN Setup message if the presentation is set to Restricted. 0 = IE aren't removed (default). 1 = IE are removed.
Add Prefix to Redirect Number **[Prefix2RedirectNumber]**	Defines a string Prefix that is added to the Redirect number received from the Tel side. This Prefix is added to the Redirect Number in the Diversion header. The valid range is an 8 character string. The default is an empty string.

5.6 Advanced Configuration

Use this menu to set the gateway's advanced configuration parameters.

Note: Parameters contained within square brackets are the names used to configure the parameters via the *ini* file.

5.6.1 Configuring the Network Settings

From the Network Settings you can:

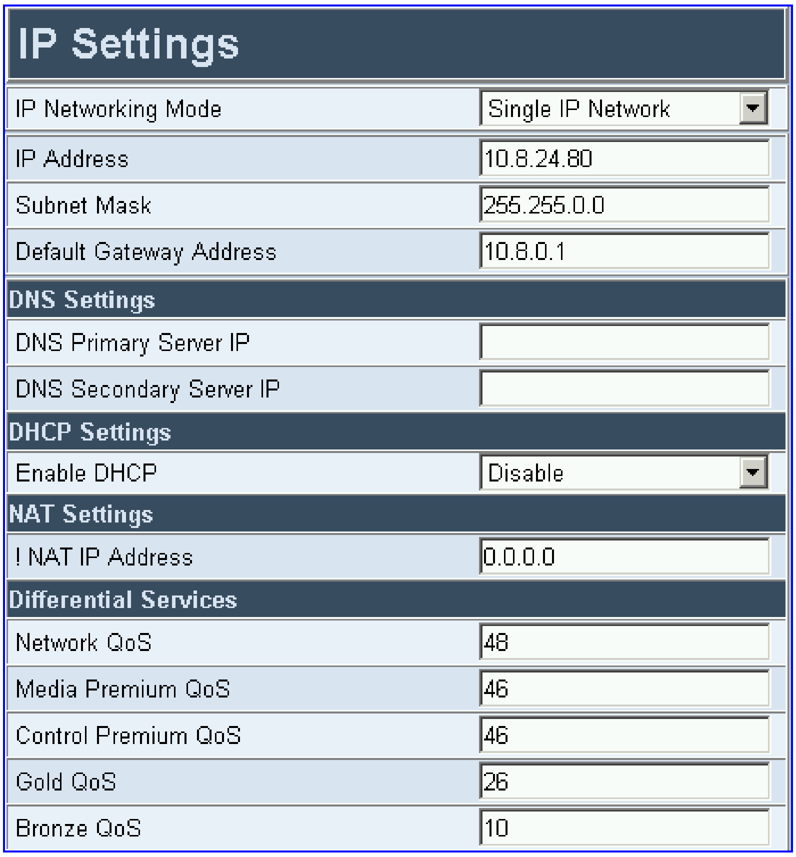
- Configure the IP Settings (refer to Section 5.6.1.1 below)
- Configure the Application Settings (refer to Section 5.6.1.2 on page 142)
- Configure the NFS Settings (refer to Section 5.6.1.3 on page 143)
- Configure the IP Routing Table (refer to Section 5.6.1.4 on page 145)
- Configure the VLAN Settings (refer to Section 5.6.1.5 on page 147)

5.6.1.1 Configuring the IP Settings

➢ To configure the IP Settings parameters, take these 4 steps:

1. Open the ‘IP Settings’ screen (**Advanced Configuration** menu > **Network Settings** > **IP Settings** option); the ‘IP Settings’ screen is displayed.

Figure 5-36: IP Settings Screen



IP Settings

IP Networking Mode	Single IP Network
IP Address	10.8.24.80
Subnet Mask	255.255.0.0
Default Gateway Address	10.8.0.1
DNS Settings	
DNS Primary Server IP	
DNS Secondary Server IP	
DHCP Settings	
Enable DHCP	Disable
NAT Settings	
! NAT IP Address	0.0.0.0
Differential Services	
Network QoS	48
Media Premium QoS	46
Control Premium QoS	46
Gold QoS	26
Bronze QoS	10

2. Configure the IP Settings according to Table 5-23.
3. Click the **Submit** button to save your changes.
4. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Table 5-23: Network Settings, IP Settings Parameters (continues on pages 139 to 140)

Parameter	Description
IP Networking Mode **[EnableMultipleIPs]**	Enables / disables the Multiple IPs mechanism. Single IP Network **[0]** (default). Multiple IP Network **[1]**. For detailed information on Multiple IPs, refer to Section 10.9 on page 308.
IP Address	IP address of the gateway. Enter the IP address in dotted format notation, for example 10.8.201.1. **Note 1:** A warning message is displayed (after pressing the button ‘Submit’) if the entered value is incorrect. **Note 2:** After changing the IP address and pressing the button ‘Submit’, a prompt appears indicating that for the change to take effect, the gateway is to be reset.

Table 5-23: Network Settings, IP Settings Parameters (continues on pages 139 to 140)

Parameter	Description
Subnet Mask	Subnet mask of the gateway. Enter the subnet mask in dotted format notation, for example 255.255.0.0 **Note 1:** A warning message is displayed (after pressing the button 'Submit') if the entered value is incorrect. **Note 2:** After changing the subnet mask and pressing the button 'Submit', a prompt appears indicating that for the change to take effect, the gateway is to be reset.
Default Gateway Address	IP address of the default gateway used by the gateway. Enter the IP address in dotted format notation, for example 10.8.0.1. **Note 1:** A warning message is displayed (after pressing the button 'Submit') if the entered value is incorrect. **Note 2:** After changing the default gateway IP address and pressing the button 'Submit', a prompt appears indicating that for the change to take effect, the gateway is to be reset. For detailed information on multiple routers support, refer to Section 10.7 on page 307.
OAM Network Settings (available only in Multiple IPs and Dual IP modes)	
IP Address **[LocalOAMIPAddress]**	The gateway's source IP address in the OAM network. The default value is 0.0.0.0.
Subnet Mask **[LocalOAMSubnetMask]**	The gateway's subnet mask in the OAM network. The default subnet mask is 0.0.0.0.
Default Gateway Address **[LocalOAMDefaultGW]**	N/A. Use the IP Routing table instead (Advanced Configuration > Network Settings).
Control Network Settings (available only in Multiple IPs and Dual IP modes)	
IP Address **[LocalControlIPAddress]**	The gateway's source IP address in the Control network. The default value is 0.0.0.0.
Subnet Mask **[LocalControlSubnetMask]**	The gateway's subnet mask in the Control network. The default subnet mask is 0.0.0.0.
Default Gateway Address **[LocalControlDefaultGW]**	N/A. Use the IP Routing table instead (Advanced Configuration > Network Settings).
Media Network Settings (available only in Multiple IPs and Dual IP modes)	
IP Address **[LocalMediaIPAddress]**	The gateway's source IP address in the Media network. The default value is 0.0.0.0.
Subnet Mask **[LocalMediaSubnetMask]**	The gateway's subnet mask in the Media network. The default subnet mask is 0.0.0.0.
Default Gateway Address **[LocalMediaDefaultGW]**	The gateway's default gateway IP address in the Media network. The default value is 0.0.0.0.
DNS Settings	
DNS Primary Server IP **[DNSPriServerIP]**	IP address of the primary DNS server. Enter the IP address in dotted format notation, for example 10.8.2.255. **Note:** To use Fully Qualified Domain Names (FQDN) in the Tel to IP Routing table, you must define this parameter.
DNS Secondary Server IP **[DNSSecServerIP]**	IP address of the second DNS server. Enter the IP address in dotted format notation, for example 10.8.2.255.

Table 5-23: Network Settings, IP Settings Parameters (continues on pages 139 to 140)

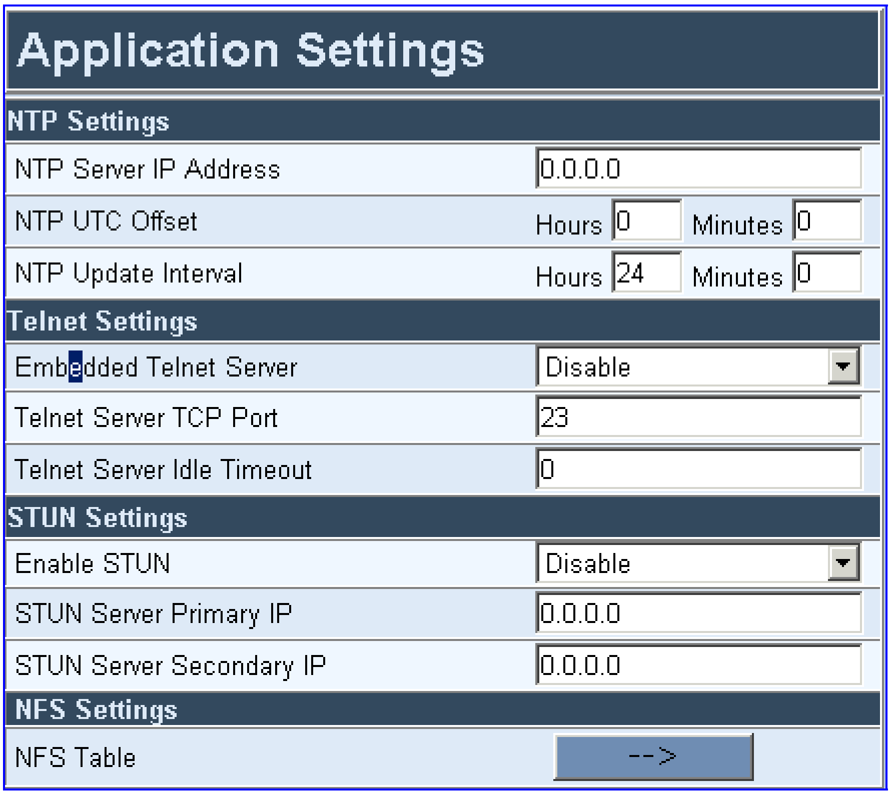
Parameter	Description
DHCP Settings	
Enable DHCP **[DHCPEnable]**	Disable **[0]** = Disable DHCP support on the gateway (default). Enable **[1]** = Enable DHCP support on the gateway. After the gateway is powered up, it attempts to communicate with a BootP server. If a BootP server is not responding and if DHCP is enabled, then the gateway attempts to get its IP address and other network parameters from the DHCP server. **Note:** After you enable the DHCP Server (from the Web browser) follow this procedure: ▪ Click the Submit button. ▪ Save the configuration. For information on how to save the configuration, refer to Section 5.9.2 on page 200. ▪ Reset the gateway *directly* (Web reset doesn't trigger the BootP/DHCP procedure and the parameter DHCPEnable reverts to '0'). Note that throughout the DHCP procedure the BootP/TFTP application must be deactivated. Otherwise, the Mediant 1000 receives a response from the BootP server instead of the DHCP server. **Note:** For additional information on DHCP, refer to Section 7.2 on page 259. ***ini* file note:** The DHCPEnable is a special 'Hidden' parameter. Once defined and saved in flash memory, its assigned value doesn't revert to its default even if the parameter doesn't appear in the *ini* file.
NAT Settings	
NAT IP Address **[StaticNatIP]**	Global gateway IP address. Define if static Network Address Translation (NAT) device is used between the gateway and the Internet.
Differential Services. For detailed information on IP QoS via Differentiated Services, refer to Section 10.9 on page 308.	
Network QoS **[NetworkServiceClassDiffServ]**	Sets the DiffServ value for Network service class content. The valid range is 0 to 56. The default value is 48.
Media Premium QoS **[PremiumServiceClassMediaDiffServ]**	Sets the DiffServ value for Premium Media service class content (only if IPDiffServ is not set in the selected IP Profile). The valid range is 0 to 56. The default value is 46. **Note:** The value for the Premium Control DiffServ is determined by (according to priority): (1) IPDiffServ value in the selected IP Profile. (2) PremiumServiceClassMediaDiffServ.
Control Premium QoS **[PremiumServiceClassControlDiffServ]**	Sets the DiffServ value for Premium Control service class content (only if ControlIPDiffserv is not set in the selected IP Profile). The valid range is 0 to 56. The default value is 46. **Note:** The value for the Premium Control DiffServ is determined by (according to priority): (1) ControlPDiffserv value in the selected IP Profile. (2) PremiumServiceClassControlDiffServ.
Gold QoS **[GoldServiceClassDiffServ]**	Sets the DiffServ value for the Gold service class content. The valid range is 0 to 56. The default value is 26.
Bronze QoS **[BronzeServiceClassDiffServ]**	Sets the DiffServ value for the Bronze service class content. The valid range is 0 to 56. The default value is 10.

5.6.1.2 Configuring the Application Settings

➢ **To configure the Application Settings parameters, take these 4 steps:**

1. Open the 'Application Settings' screen (**Advanced Configuration** menu > **Network Settings** > **Application Settings** option); the 'Application Settings' screen is displayed.

Figure 5-37: Application Settings Screen



Application Settings

NTP Settings

NTP Server IP Address	0.0.0.0
NTP UTC Offset	Hours 0 Minutes 0
NTP Update Interval	Hours 24 Minutes 0

Telnet Settings

Embedded Telnet Server	Disable
Telnet Server TCP Port	23
Telnet Server Idle Timeout	0

STUN Settings

Enable STUN	Disable
STUN Server Primary IP	0.0.0.0
STUN Server Secondary IP	0.0.0.0

NFS Settings

NFS Table	-->

2. Configure the Application Settings according to Table 5-24.
3. Click the **Submit** button to save your changes.
4. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Table 5-24: Network Settings, Application Settings Parameters (continues on pages 142 to 143)

Parameter	Description
NTP Settings	
For detailed information on NTP, refer to Section 10.8 on page 308.	
NTP Server IP Address **[NTPServerIP]**	IP address (in dotted format notation) of the NTP server. The default IP address is 0.0.0.0 (the internal NTP client is disabled).
NTP UTC Offset **[NTPServerUTCOffset]**	Defines the UTC (Universal Time Coordinate) offset (in seconds) from the NTP server. The default offset is 0. The offset range is –43200 to 43200 seconds.

Table 5-24: Network Settings, Application Settings Parameters (continues on pages 142 to 143)

Parameter	Description
NTP Update Interval **[NTPUpdateInterval]**	Defines the time interval (in seconds) the NTP client requests for a time update. The default interval is 86400 seconds (24 hours). The range is 0 to 214783647 seconds. **Note:** It isn't recommended to be set beyond one month (2592000 seconds).
Telnet Settings	
Embedded Telnet Server **[TelnetServerEnable]**	Enables or disables the embedded Telnet server. Telnet is disabled by default for security reasons. Disable **[0]** (default). Enable (Unsecured) **[1]**. Enable Secured (SSL) **[2]**.
Telnet Server TCP Port **[TelnetServerPort]**	Defines the port number for the embedded Telnet server. The valid range = valid port numbers. The default port is 23.
Telnet Server Idle Timeout **[TelnetServerIdleDisconnect]**	Sets the timeout for disconnection of an idle Telnet session (in minutes). When set to zero, idle sessions are not disconnected. The valid range is any value. The default value is 0.
STUN Settings	
Enable STUN **[EnableSTUN]**	Disable **[0]** = STUN protocol is disabled (default). Enable **[1]** = STUN protocol is enabled. When enabled, the gateway functions as a STUN client and communicates with a STUN server located in the open internet. STUN is used to discover whether the gateway is located behind a NAT and the type of that NAT. In addition, it is used to determine the IP addresses and port numbers that the NAT assigns to outgoing signaling messages (using SIP) and media streams (using RTP, RTCP and T.38). STUN works with many existing NAT types, and does not require any special behavior from them. This parameter cannot be changed on-the-fly and requires a gateway reset. For detailed information on STUN, refer to Section 10.3.1 on page 303. Note: For defining the STUN server domain name, use the *ini* file parameter STUNServerDomainName (refer to Section 6.5.1 on page 215).
STUN Server Primary IP **[STUNServerPrimaryIP]**	Defines the IP address of the primary STUN server. The valid range is the legal IP addresses. The default value is 0.0.0.0.
STUN Server Secondary IP **[STUNServerSecondaryIP]**	Defines the IP address of the secondary STUN server. The valid range is the legal IP addresses. The default value is 0.0.0.0.
NFS Settings	
NFS Table	For detailed information on configuring the NFS table, refer to Section 5.6.1.3 on page 143.

5.6.1.3 Configuring the NFS Settings

Network File System (NFS) enables the Mediant 1000 to access a remote server's shared files and directories and to handle them as if they're located locally. A file system, the NFS is independent of machine types, OSs, and network architectures. Up to five different NFS file systems can be configured.

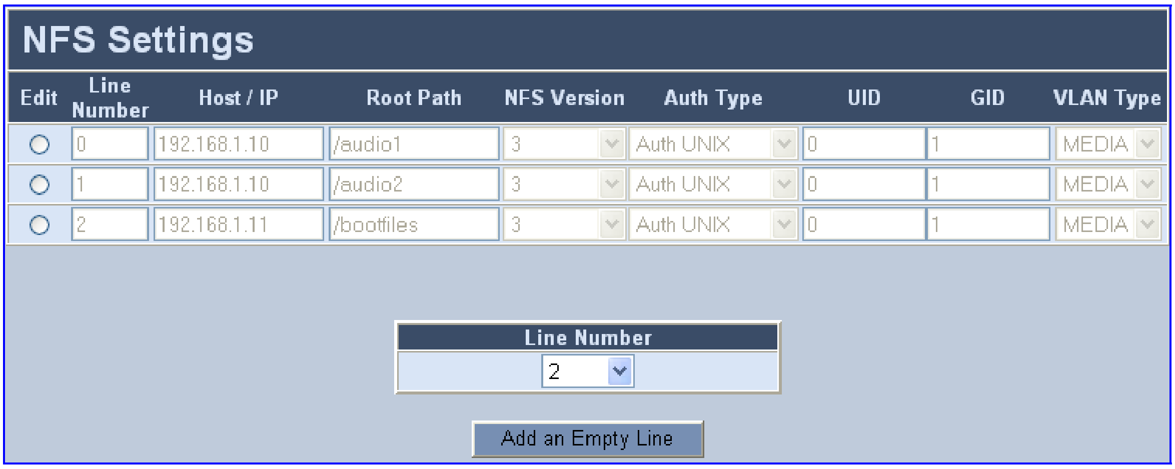
NFS is utilized by the Mediant 1000 to load the *cmp*, *ini* and configuration files via the Automatic Update mechanism (refer to Section 12.3 on page 320).

Note that an NFS file server can share multiple file systems. There must be a separate row for each remote file system shared by the NFS file server that needs to be accessed by the Mediant 1000.

➢ To configure the NFS Settings parameters, take these 7 steps:

1. Open the ‘Application Settings’ screen (**Advanced Configuration** menu > **Network Settings** > **Application Settings** option); the ‘Application Settings’ screen is displayed (Figure 5-37).
2. Open the NFS Table screen by clicking the arrow sign (-->) to the right of the NFS label; the NFS Table screen is displayed (Figure 5-38).

Figure 5-38: NFS Settings Table Screen



NFS Settings

Edit	Line Number	Host / IP	Root Path	NFS Version	Auth Type	UID	GID	VLAN Type
○	0	192.168.1.10	/audio1	3	Auth UNIX	0	1	MEDIA
○	1	192.168.1.10	/audio2	3	Auth UNIX	0	1	MEDIA
○	2	192.168.1.11	/bootfiles	3	Auth UNIX	0	1	MEDIA

Line Number: 2

3. To add a remote NFS file system, select an available line number from the 'Line Number' drop-down list.
4. Click the **Add an Empty Line** button; an empty line appears.
5. Configure the NFS Settings according to Table 5-25.
6. Click the **Apply New Settings** button; the remote NFS file system is mounted immediately. Check the Syslog server for the ‘NFS mount was successful’ message.
7. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Note: To avoid terminating calls in progress, a row must not be deleted or modified while the board is currently accessing files on that remote NFS file system.

➢ To delete a remote NFS file system, take these 3 steps:

1. Select the **Edit** radio button for the row to be deleted.
2. Click the **Delete Line** button; the row is deleted.
3. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

➢ To modify an existing remote NFS file system, take these 4 steps:

1. Select the **Edit** radio button for the row to be modified.
2. Change the values on the selected row according to your requirements.
3. Click the **Apply New Settings** button; the remote NFS file system is mounted using the new settings. Check the Syslog server for the ‘NFS mount was successful’ message.
4. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Table 5-25: Network Settings, NFS Settings Parameters

Parameter	Description
Line Number **[NFSServers_Index]**	The row index of the remote file system. The valid range is 0 to 4.
Host / IP **[NFSServers_HostOrIP]**	The domain name or IP address of the NFS server. If a domain name is provided, a DNS server must be configured.
Root Path **[NFSServers_RootPath]**	Path to the root of the remote file system. In the format: '/' + [path] For example, /audio
The combination of Host / IP and Root Path must be unique for each row in the table. For example, there must be only one row in the table with a Host / IP of 192.168.1.1 and Root Path of /audio.	
NFS Version **[NFSServers_NfsVersion]**	NFS version to use with the remote file system, 2 or 3 (default).
Auth Type **[NFSServers_AuthType]**	Identifies the authentication method used with the remote file system. Auth NULL **[0]**. Auth UNIX **[1]** (default).
UID **[NFSServers_UID]**	User ID used in authentication if using Auth UNIX. The valid range is 0 to 65537. The default is 0.
GID **[NFSServers_GID]**	Group ID used in authentication if using Auth UNIX. The valid range is 0 to 65537. The default is 1
VLAN Type **[NFSServers_VlanType]**	The VLAN, OAM **[0]** or Media **[1]**, to use when accessing the remote file system. The default is to use the media VLAN. This parameter applies only if VLANs are enabled or if Multiple IPs is configured (refer to Section 10.9 on page 308).

Figure 5-39 below shows an example of an NFS table definition via *ini* file using parameter tables (for information on parameter tables, refer to Section 12.5 on page 325).

Figure 5-39: NFS *ini* File Example

```
[ NFSServers ]
FORMAT NFSServers Index = NFSServers HostOrIP,
NFSServers RootPath, NFSServers NfsVersion, NFSServers AuthType,
NFSServers_UID, NFSServers_GID, NFSServers_VlanType;
NFSServers 1 = 101.1.13, /audio1, 3, 1, 0, 1, 1;
[ \NFSServers ]
```

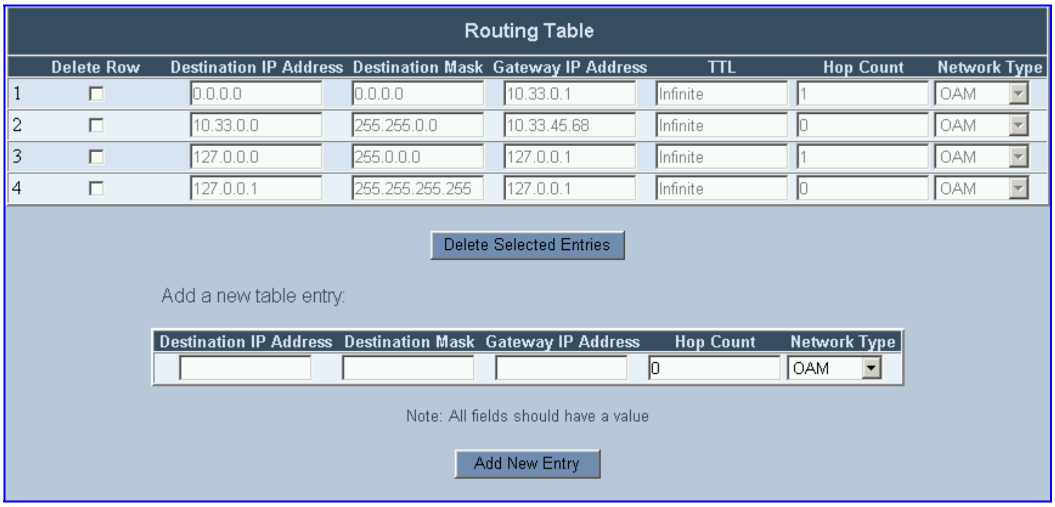
5.6.1.4 Configuring the IP Routing Table

The IP routing table is used by the gateway to determine IP routing rules. It can be used, for example, to define static routing rules for the OAM and Control networks since a default gateway isn't supported for these networks (refer to Section 10.10.1 on page 309). Before sending an IP packet, the gateway searches this table for an entry that matches the requested destination host / network. If such entry is found, the gateway sends the packet to the indicated router. If no explicit entry is found, the packet is sent to the default gateway (configured in Network Settings>IP Settings screen). Up to 50 routing entries are available.

➢ To configure the IP Routing table, take these 3 steps:

1. Open the ‘IP Routing Table’ screen (**Advanced Configuration** menu > **Network Settings** > **Routing Table** option); the ‘IP Routing Table’ screen is displayed.

Figure 1-3: IP Routing Table Screen



Routing Table

	Delete Row	Destination IP Address	Destination Mask	Gateway IP Address	TTL	Hop Count	Network Type
1	□	0.0.0.0	0.0.0.0	10.33.0.1	Infinite	1	OAM
2	□	10.33.0.0	255.255.0.0	10.33.45.68	Infinite	0	OAM
3	□	127.0.0.0	255.0.0.0	127.0.0.1	Infinite	1	OAM
4	□	127.0.0.1	255.255.255.255	127.0.0.1	Infinite	0	OAM

Add a new table entry:

Destination IP Address	Destination Mask	Gateway IP Address	Hop Count	Network Type
			0	OAM

Note: All fields should have a value

2. Use the ‘Add a new table entry’ pane to add a new routing rule. Each field in the IP routing table is described in Table 5-26.
3. Click the button **Add New Entry**; the new routing rule is added to the IP routing table.

Table 5-26: IP Routing Table Column Description

Column Name **[ini File Parameter Name]**	**Description**
Delete Row	To delete IP routing rules from the IP Routing Table, check the Delete Row checkbox in the rows of the routing rules you want to delete and click the button **Delete Selected Entries**; the routing rules are removed from the table.
Destination IP Address **[RoutingTableDestinationsColumn]**	Specifies the IP address of the destination host / network.
Destination Mask **[RoutingTableDestinationMasksColumn]**	Specifies the subnet mask of the destination host / network.
The address of the host / network you want to reach is determined by an AND operation that is applied on the fields ‘Destination IP Address’ and ‘Destination Mask’. For example: To reach the network 10.8.x.x, enter 10.8.0.0 in the field ‘Destination IP Address’ and 255.255.0.0 in the field ‘Destination Mask’. As a result of the AND operation, the value of the last two octets in the field ‘Destination IP Address’ is ignored. To reach a specific host, enter its IP address in the field ‘Destination IP Address’ and 255.255.255.255 in the field ‘Destination Mask’.	
Gateway IP Address **[RoutingTableGatewaysColumn]**	Specifies the IP address of the router to which the packets are sent if their destination matches the rules in the adjacent columns.
TTL	A read-only field that indicates the time period for which the specific routing rule is valid. The lifetime of a static route is infinite.
Hop Count **[RoutingTableHopsCountColumn]**	The maximum number of allowed routers between the gateway and destination.

Table 5-26: IP Routing Table Column Description

Column Name **[ini File Parameter Name]**	**Description**
Network Type **[RoutingTableInterfacesColumn]**	Specifies the network type the routing rule is applied to. OAM **[0]** (default). Control **[1]**. Media **[2]**. For detailed information on the network types, refer to Section 10.9 on page 308.

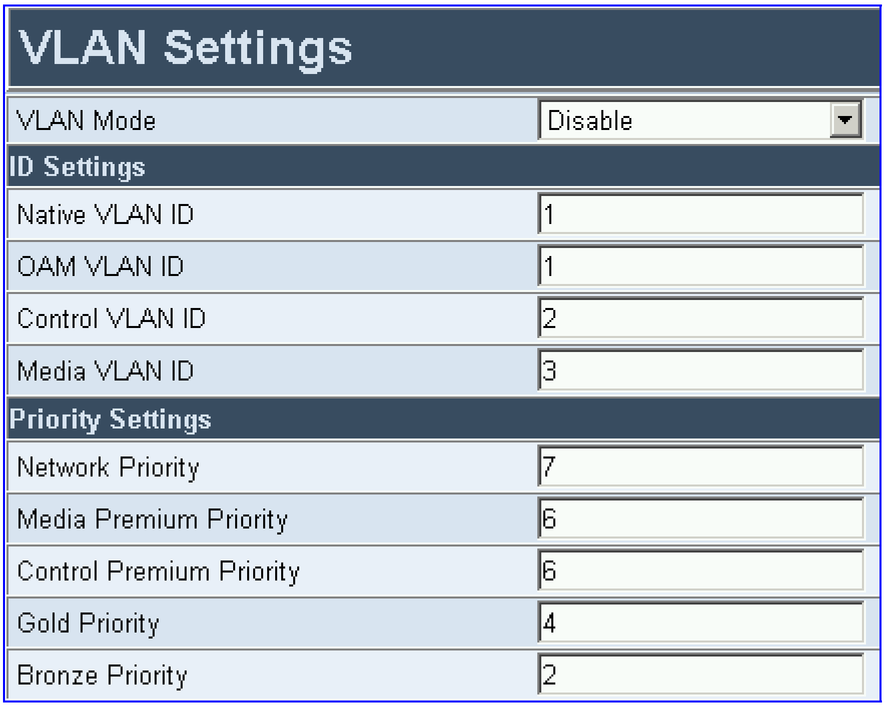
5.6.1.5 Configuring the VLAN Settings

For detailed information on the Mediant 1000 VLAN implementation, refer to Section 10.9 on page 308.

➢ **To configure the VLAN Settings parameters, take these 4 steps:**

1. Open the 'VLAN Settings' screen (**Advanced Configuration** menu > **Network Settings** > **VLAN Settings** option); the 'VLAN Settings' screen is displayed.

Figure 5-40: VLAN Settings Screen



VLAN Settings

VLAN Mode: Disable

ID Settings

Native VLAN ID: 1

OAM VLAN ID: 1

Control VLAN ID: 2

Media VLAN ID: 3

Priority Settings

Network Priority: 7

Media Premium Priority: 6

Control Premium Priority: 6

Gold Priority: 4

Bronze Priority: 2

2. Configure the VLAN Settings according to Table 5-27.
3. Click the **Submit** button to save your changes.
4. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Table 5-27: Network Settings, VLAN Settings Parameters (continues on pages 148 to 148)

Parameter	Description
VLAN Mode **[VlANMode]**	Sets the VLAN functionality. Disable **[0]** (default). Enable **[1]**. PassThrough **[2]** = N/A. **Note:** This parameter cannot be changed on-the-fly and requires a gateway reset.
IP Settings	
Native VLAN ID **[VLANNativeVlanID]**	Sets the native VLAN identifier (PVID, Port VLAN ID). The valid range is 1 to 4094. The default value is 1.
OAM VLAN ID **[VLANOamVlanID]**	Sets the OAM (Operation, Administration and Management) VLAN identifier. The valid range is 1 to 4094. The default value is 1.
Control VLAN ID **[VLANControlVlanID]**	Sets the control VLAN identifier. The valid range is 1 to 4094. The default value is 2.
Media VLAN ID **[VLANMediaVlanID]**	Sets the media VLAN identifier. The valid range is 1 to 4094. The default value is 3.
Priority Settings	
Network Priority **[VLANNetworkServiceClassPriority]**	Sets the priority for Network service class content. The valid range is 0 to 7. The default value is 7.
Media Premium Priority **[VLANPremiumServiceClassMediaPriority]**	Sets the priority for the Premium service class content and media traffic. The valid range is 0 to 7. The default value is 6.
Control Premium Priority **[VLANPremiumServiceClassControlPriority]**	Sets the priority for the Premium service class content and control traffic. The valid range is 0 to 7. The default value is 6.
Gold Priority **[VLANGoldServiceClassPriority]**	Sets the priority for the Gold service class content. The valid range is 0 to 7. The default value is 4.
Bronze Priority **[VLANBronzeServiceClassPriority]**	Sets the priority for the Bronze service class content. The valid range is 0 to 7. The default value is 2.

5.6.2 Configuring the Media Settings

Use these menus to set the gateway's channel parameters. These parameters are applied to all Mediant 1000 channels.

From the Media Settings page you can define:

- Voice Settings (refer to Section 5.6.2.1).
- Fax / Modem / CID Settings (refer to Section 5.6.2.2 on page 151).
- RTP/RTCP Settings (refer to Section 5.6.2.3 on page 154).
- Hook-Flash Settings (refer to Section 5.6.2.4 on page 157).
- General Media Settings (refer to Section 5.6.2.5 on page 158).

Notes:

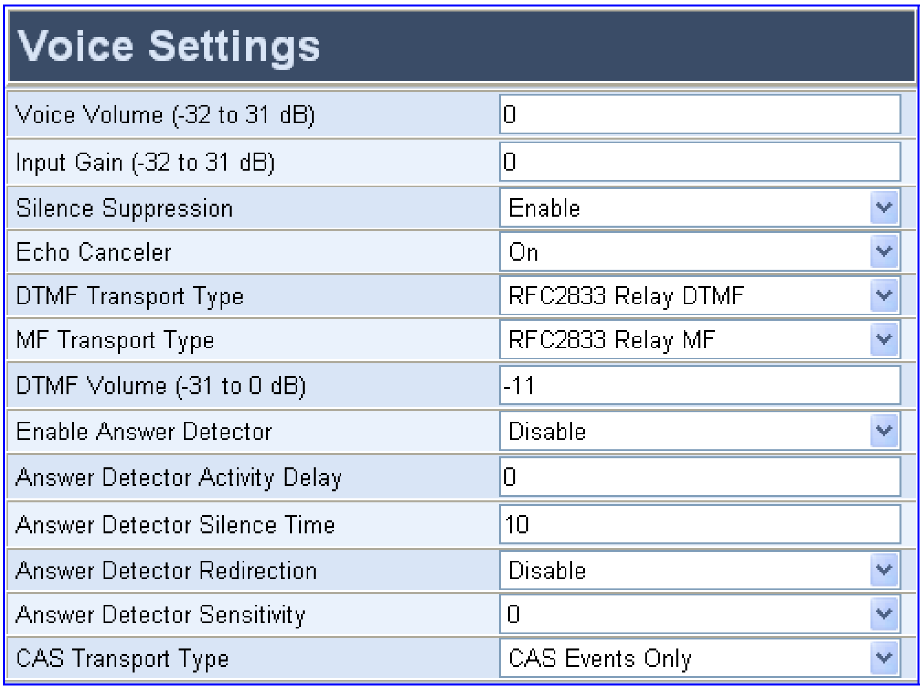
- Parameters contained within square brackets are the names used to configure the parameters via the *ini* file.
- Channel parameters are changeable on-the-fly. Changes take effect from next call.
- Several Channels Settings parameters can be configured per call using profiles (refer to Section 5.5.5 on page 115).
- The parameter 'Fax Transport Mode' (Fax / Modem / CID Settings screen) is overridden by the parameter 'IsFaxUsed'.

5.6.2.1 Configuring the Voice Settings

➢ **To configure the Voice Settings parameters, take these 4 steps:**

1. Open the 'Voice Settings' screen (**Advanced Configuration** menu > **Media Settings** > **Voice Settings** option); the 'Voice Settings' screen is displayed.

Figure 5-41: Voice Settings Screen



Voice Settings

Voice Volume (-32 to 31 dB)	0
Input Gain (-32 to 31 dB)	0
Silence Suppression	Enable
Echo Canceler	On
DTMF Transport Type	RFC2833 Relay DTMF
MF Transport Type	RFC2833 Relay MF
DTMF Volume (-31 to 0 dB)	-11
Enable Answer Detector	Disable
Answer Detector Activity Delay	0
Answer Detector Silence Time	10
Answer Detector Redirection	Disable
Answer Detector Sensitivity	0
CAS Transport Type	CAS Events Only

2. Configure the Voice Settings according to Table 5-28.
3. Click the **Submit** button to save your changes.
4. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Table 5-28: Media Settings, Voice Settings Parameters

Parameter	Description
Voice Volume **[VoiceVolume]**	Voice gain control in dB. This parameter sets the level for the transmitted (IP→PSTN/Tel) signal. The valid range is -32 to 31 dB. The default value is 0 dB.
Input Gain **[InputGain]**	PCM input gain control in dB. This parameter sets the level for the received (Tel/PSTN→IP) signal. The valid range is -32 to 31 dB. The default value is 0 dB. **Note:** This parameter is intended for advanced users. Changing it affects other gateway functionalities.

Table 5-28: Media Settings, Voice Settings Parameters

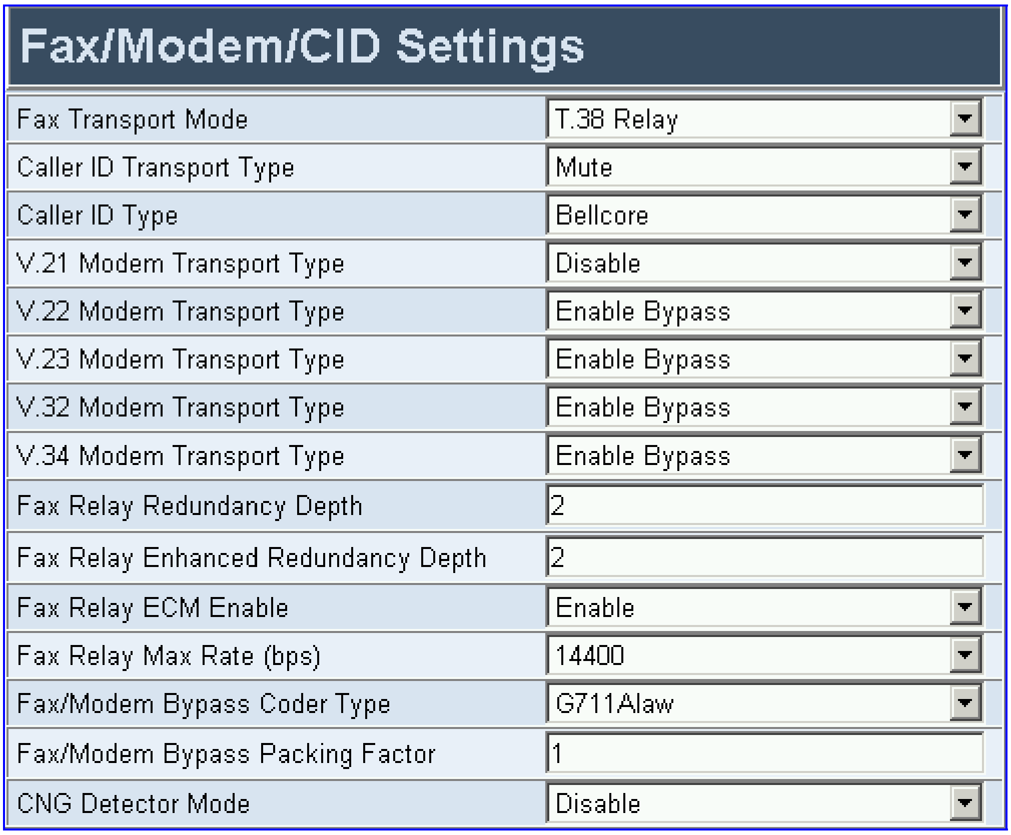
Parameter	Description
Silence Suppression **[EnableSilenceCompression]**	Silence Suppression is a method conserving bandwidth on VoIP calls by not sending packets when silence is detected. Disable **[0]** = Silence Suppression disabled (default). Enable **[1]** = Silence Suppression enabled. Enable without adaptation **[2]** = A single silence packet is sent during silence period (applicable only to G.729). **Note:** If the selected coder is G.729, the following rules determine the value of the 'annexb' parameter of the fmtp attribute in the SDP. EnableSilenceCompression = 0 → 'annexb=no'. EnableSilenceCompression = 1 → 'annexb=yes'. EnableSilenceCompression = 2 and IsCiscoSCEMode = 0 → 'annexb=yes'. EnableSilenceCompression = 2 and IsCiscoSCEMode = 1 → 'annexb=no'.
Echo Canceler **[EnableEchoCanceller]** The parameter **ECE** is used to maintain backward compatibility.	Off **[0]** = Echo Canceler disabled. On **[1]** = Echo Canceler enabled (default).
DTMF Transport Type **[DTMFTransportType]**	DTMF Mute **[0]** = Erase digits from voice stream, do not relay to remote. Transparent DTMF **[2]** = Digits remain in voice stream. RFC 2833 Relay DTMF **[3]** = Erase digits from voice stream, relay to remote according to RFC 2833 (default). **Note:** This parameter is automatically updated if one of the following parameters is configured: TxDTMFOption or RxDTMFOption.
MF Transport Type **[MFTransportType]**	N/A.
DTMF Volume (-31 to 0 dB) **[DTMFVolume]**	DTMF gain control value in dB (to the TDM side). The valid range is -31 to 0 dB.The default value is -11 dB.
Enable Answer Detector **[EnableAnswerDetector]**	N/A.
Answer Detector Activity Delay **[AnswerDetectorActivityDelay]**	N/A.
Answer Detector Silence Time **[AnswerDetectorSilenceTime]**	N/A.
Answer Detector Redirection **[AnswerDetectorRedirection]**	N/A.
Answer Detector Sensitivity **[AnswerDetectorSensitivity]**	Determines the Answer Detector sensitivity. The range is 0 (most sensitive) to 2 (least sensitive). The default is 0.
CAS Transport Type **[CASTransportType]**	0 = Disable CAS relay (default). 1 = Enable CAS relay mode using RFC 2833. The CAS relay mode can be used with the TDM tunneling feature to enable tunneling over IP for both voice and CAS signaling bearers.

5.6.2.2 Configuring the Fax / Modem / CID Settings

➢ **To configure the Fax / Modem / CID Settings parameters, take these 4 steps:**

1. Open the 'Fax / Modem / CID Settings' screen (**Advanced Configuration** menu > **Media Settings** > **Fax / Modem / CID Settings** option); the 'Fax / Modem / CID Settings' screen is displayed.

Figure 5-42: Fax / Modem / CID Settings Screen



Fax/Modem/CID Settings

Fax Transport Mode	T.38 Relay
Caller ID Transport Type	Mute
Caller ID Type	Bellcore
V.21 Modem Transport Type	Disable
V.22 Modem Transport Type	Enable Bypass
V.23 Modem Transport Type	Enable Bypass
V.32 Modem Transport Type	Enable Bypass
V.34 Modem Transport Type	Enable Bypass
Fax Relay Redundancy Depth	2
Fax Relay Enhanced Redundancy Depth	2
Fax Relay ECM Enable	Enable
Fax Relay Max Rate (bps)	14400
Fax/Modem Bypass Coder Type	G711Alaw
Fax/Modem Bypass Packing Factor	1
CNG Detector Mode	Disable

2. Configure the Fax / Modem / CID Settings according to Table 5-29.
3. Click the **Submit** button to save your changes.
4. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Table 5-29: Media Settings, Fax/Modem/CID Parameters (continues on pages 152 to 153)

Parameter	Description
Fax Transport Mode **[FaxTransportMode]**	Fax Transport Mode that the gateway uses. You can select: Disable **[0]** (transparent mode). T.38 Relay **[1]** (default). Bypass **[2]**. Events Only **[3]**. **Note:** If parameter IsFaxUsed = 1, then FaxTransportMode is always set to 1 (T.38 relay).
Caller ID Transport Type **[CallerIDTransportType]**	N/A.
Caller ID Type **[CallerIDType]**	Defines one of the following standards for detection (FXO) and generation (FXS) of Caller ID and detection (FXO) of MWI (when specified) signals. Bellcore **[0]** (Caller ID and MWI) (default). ETSI **[1]** (Caller ID and MWI) NTT **[2]**. British **[4]** DTMF ETSI **[16]** Denmark **[17]** (Caller ID and MWI) India **[18]** Brazil **[19]** **Note 1:** The Caller ID signals are generated/detected between the first and the second rings. **Note 2:** To select the Bellcore Caller ID sub standard, use the parameter 'BellcoreCallerIDTypeOneSubStandard'. To select the ETSI Caller ID sub standard, use the parameter 'ETSICallerIDTypeOneSubStandard'. **Note 3:** To select the Bellcore MWI sub standard, use the parameter 'BellcoreVMWITypeOneStandard'. To select the ETSI MWI sub standard, use the parameter 'ETSIVMWITypeOneStandard'.
V.21 Modem Transport Type **[V21ModemTransportType]**	V.21 Modem Transport Type that the gateway uses. You can select: 0 = Disable (Transparent) -- default 1 = Enable Relay – N/A 2 = Enable Bypass 3 = Events Only (Transparent with Events)
V.22 Modem Transport Type **[V22ModemTransportType]**	V.22 Modem Transport Type that the gateway uses. You can select: 0 = Disable (Transparent) 1 = Enable Relay -- N/A 2 = Enable Bypass -- default 3 = Events Only (Transparent with Events)
V.23 Modem Transport Type **[V23ModemTransportType]**	V.23 Modem Transport Type that the gateway uses. You can select: 0 = Disable (Transparent) 1 = Enable Relay -- N/A 2 = Enable Bypass -- default 3 = Events Only (Transparent with Events)
V.32 Modem Transport Type **[V32ModemTransportType]**	V.32 Modem Transport Type that the gateway uses. You can select: 0 = Disable (Transparent) 1 = Enable Relay -- N/A 2 = Enable Bypass -- default 3 = Events Only (Transparent with Events) **Note:** This option applies to V.32 and V.32bis modems.

Table 5-29: Media Settings, Fax/Modem/CID Parameters (continues on pages 152 to 153)

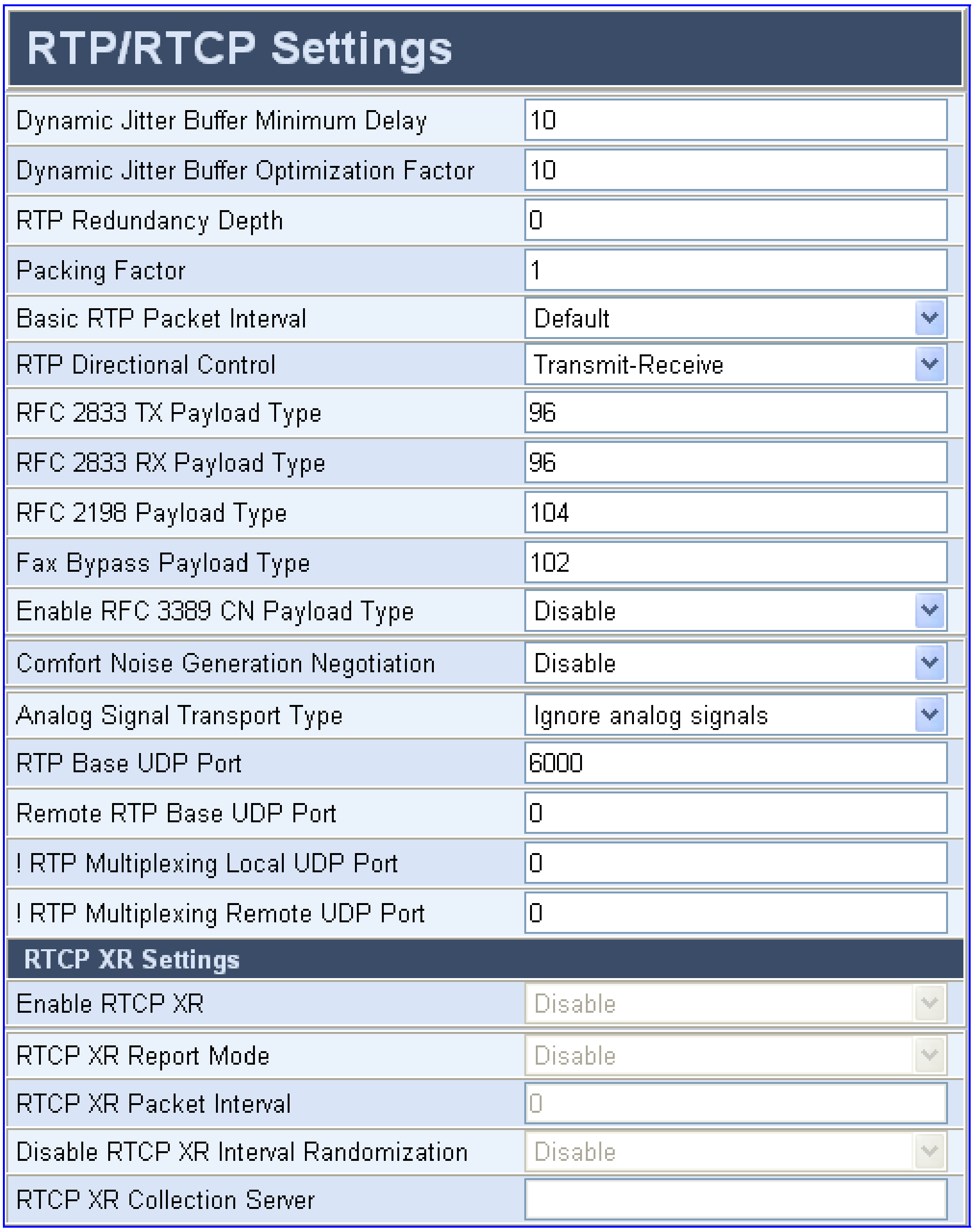
Parameter	Description
V.34 Modem Transport Type **[V34ModemTransportType]**	V.90 / V.34 Modem Transport Type that the gateway uses. You can select: 0 = Disable (Transparent) 1 = Enable Relay -- N/A 2 = Enable Bypass -- default 3 = Events Only (Transparent with Events)
Fax Relay Redundancy Depth **[FaxRelayRedundancyDepth]**	Number of times that each fax relay payload is retransmitted to the network. The valid range is 0 to 2. The default value is 0.
Fax Relay Enhanced Redundancy Depth **[FaxRelayEnhancedRedundancyDepth]**	Number of times that control packets are retransmitted when using the T.38 standard. The valid range is 0 to 4. The default value is 2.
Fax Relay ECM Enable **[FaxRelayECMEnable]**	Disable **[0]** = Error Correction Mode (ECM) mode is not used during fax relay. Enable **[1]** = ECM mode is used during fax relay (default).
Fax Relay Max Rate (bps) **[FaxRelayMaxRate]**	Maximum rate (in bps), at which fax relay messages are transmitted (outgoing calls). You can select: 2400 **[0]** = 2.4 kbps. 4800 **[1]** = 4.8 kbps. 7200 **[2]** = 7.2 kbps. 9600 **[3]** = 9.6 kbps. 12000 **[4]** = 12.0 kbps. 14400 **[5]** = 14.4 kbps (default). **Note:** The rate is negotiated between the sides, i.e., the gateway adapts to the capabilities of the remote side.
Fax/Modem Bypass Coder Type **[FaxModemBypassCoderType]**	Coder the gateway uses when performing fax/modem bypass. Usually, high-bit-rate coders such as G.711 should be used. You can select: G.711 A-law 64 **[0]** (default). G.711 μ-law **[1]**.
Fax/Modem Bypass Packing Factor **[FaxModemBypassM]**	Number of (20 msec) coder payloads that are used to generate a fax/modem bypass packet. The valid range is 1, 2 or 3 coder payloads. The default value is 1 coder payload.
CNG Detector Mode **[CNGDetectorMode]**	**[0]** = Disable (default). Don't detect CNG. **[1]** = Relay. CNG is detected on the originating side. CNG packets are sent to the remote side according to T.38 (if IsFaxUsed=1) and the fax session is started. **[2]** = Events Only. CNG is detected on the originating side. The CNG signal passes transparently to the remote side and fax session is started. Usually T.38 fax session starts when the 'preamble' signal is detected by the answering side. Some SIP gateways don't support the detection of this fax signal on the answering side, thus, for these cases it is possible to configure the gateways to start the T.38 fax session when the CNG tone is detected by the originating side. However, this mode is not recommended.

5.6.2.3 Configuring the RTP / RTCP Settings

➢ **To configure the RTP / RTCP Settings parameters, take these 4 steps:**

1. Open the 'RTP / RTCP Settings' screen (**Advanced Configuration** menu > **Media Settings** > **RTP / RTCP Settings** option); the 'RTP / RTCP Settings' screen is displayed.

Figure 5-43: RTP / RTCP Settings Screen



RTP/RTCP Settings

Dynamic Jitter Buffer Minimum Delay	10
Dynamic Jitter Buffer Optimization Factor	10
RTP Redundancy Depth	0
Packing Factor	1
Basic RTP Packet Interval	Default
RTP Directional Control	Transmit-Receive
RFC 2833 TX Payload Type	96
RFC 2833 RX Payload Type	96
RFC 2198 Payload Type	104
Fax Bypass Payload Type	102
Enable RFC 3389 CN Payload Type	Disable
Comfort Noise Generation Negotiation	Disable
Analog Signal Transport Type	Ignore analog signals
RTP Base UDP Port	6000
Remote RTP Base UDP Port	0
! RTP Multiplexing Local UDP Port	0
! RTP Multiplexing Remote UDP Port	0
RTCP XR Settings	
Enable RTCP XR	Disable
RTCP XR Report Mode	Disable
RTCP XR Packet Interval	0
Disable RTCP XR Interval Randomization	Disable
RTCP XR Collection Server	

2. Configure the RTP / RTCP Settings according to Table 5-30.
3. Click the **Submit** button to save your changes.
4. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Table 5-30: Media Settings, RTP / RTCP Parameters (continues on pages 155 to 156)

Parameter	Description
Dynamic Jitter Buffer Minimum Delay **[DJBufMinDelay]**	Minimum delay for the Dynamic Jitter Buffer. The valid range is 0 to 150 milliseconds. The default delay is 10 milliseconds. **Note:** For more information on the Jitter Buffer, refer to Section 9.7 on page 283.
Dynamic Jitter Buffer Optimization Factor **[DJBufOptFactor]**	Dynamic Jitter Buffer frame error / delay optimization factor. The valid range is 0 to 13. The default factor is 10. **Note 1:** Set to 13 for data (fax and modem) calls. **Note 2:** For more information on the Jitter Buffer, refer to Section 9.7 on page 283.
RTP Redundancy Depth **[RTPRedundancyDepth]**	**[0]** = Disable the generation of redundant packets (default). **[1]** = Enable the generation of RFC 2198 redundancy packets.
Packing Factor **[RTPPackingFactor]**	N/A. Controlled internally by the gateway according to the selected coder.
Basic RTP Packet Interval **[BasicRTPPacketInterval]**	N/A. Controlled internally by the gateway according to the selected coder. **Note:** This parameter should not be used. Use the 'Coders' screen under 'Protocol Definition' instead.
RTP Directional Control **[RTPDirectionControl]**	N/A. Controlled internally by the gateway according to the selected coder.
RFC 2833 TX Payload Type **[RFC2833TxPayloadType]**	N/A. Use the *ini* file parameter RFC2833PayloadType instead.
RFC 2833 RX Payload Type **[RFC2833RxPayloadType]**	N/A. Use the *ini* file parameter RFC2833PayloadType instead.
RFC 2198 Payload Type **[RFC2198PayloadType]**	RTP redundancy packet payload type, according to RFC 2198. The range is 96-127. The default is 104. Applicable if 'RTP Redundancy Depth=1'
Fax Bypass Payload Type **[FaxBypassPayloadType]**	Determines the fax bypass RTP dynamic payload type. The valid range is 96 to 120. The default value is 102.
Enable RFC 3389 CN Payload Type **[EnableStandardSIDPayloadType]**	Determines whether Silence Indicator (SID) packets that are sent and received are according to RFC 3389. Disable **[0]** = G.711 SID packets are sent in a proprietary method (default). Enable **[1]** = SID (comfort noise) packets are sent with the RTP SID payload type according to RFC 3389. Applicable to G.711 and G.726 coders.
Comfort Noise Generation Negotiation **[ComfortNoiseNegotiation]**	Enables negotiation and usage of Comfort Noise (CN). Valid options include: 0 = Disable (default) 1 = Enable Comfort Noise negotiation The use of CN is indicated by including a payload type for CN on the media description line of the SDP. The gateway can use CN with a codec whose RTP timestamp clock rate is 8,000 Hz (G.711/G.726). The static payload type 13 is used. The use of CN is negotiated between sides; therefore, if the remote side doesn't support CN, it is not used. **Note:** Silence Suppression must be enabled to generate CN.
Analog Signal Transport Type **[AnalogSignalTransportType]**	This parameter is obsolete; use instead the parameter HookFlashOption.

Table 5-30: Media Settings, RTP / RTCP Parameters (continues on pages 155 to 156)

Parameter	Description
RTP Base UDP Port **[BaseUDPPort]**	Lower boundary of UDP port used for RTP, RTCP (Real-Time Control Protocol) (RTP port + 1) and T.38 (RTP port + 2). The upper boundary is the Base UDP Port + 10 * (number of gateway's channels). The range of possible UDP ports is 6,000 to 64,000. The default base UDP port is 6000. For example: If the Base UDP Port is set to 6000 (the default) then: The first channel uses the following ports: RTP 6000, RTCP 6001 and T.38 6002, the second channel uses: RTP 6010, RTCP 6011 and T.38 6012, etc. **Note:** If RTP Base UDP Port is not a factor of 10, the following message is generated: 'invalid local RTP port'. For detailed information on the default RTP/RTCP/T.38 port allocation, refer to the Section E.3 on page 426.
Remote RTP Base UDP Port **[RemoteBaseUDPPort]**	Determines the lower boundary of UDP ports used for RTP, RTCP and T.38 by a remote gateway. If this parameter is set to a non-zero value, ThroughPacket™ is enabled. Note that the value of 'RemoteBaseUDPPort' on the local gateway must equal the value of 'BaseUDPPort' of the remote gateway. The gateway uses these parameters to identify and distribute the payloads from the received multiplexed IP packet to the relevant channels. The valid range is the range of possible UDP ports: 6,000 to 64,000. The default value is 0 (ThroughPacket™ is disabled). For detailed information on ThroughPacket™, refer to Section 9.5 on page 281. **Note:** To enable ThroughPacket™ the parameters 'L1L1ComplexTxUDPPort' and 'L1L1ComplexRxUDPPort' must be set to a non-zero value.
RTP Multiplexing Local UDP Port **[L1L1ComplexTxUDPPort]**	Determines the local UDP port used for outgoing multiplexed RTP packets (applies to the ThroughPacket™ mechanism). The valid range is the range of possible UDP ports: 6,000 to 64,000. The default value is 0 (ThroughPacket™ is disabled). This parameter cannot be changed on-the-fly and requires a gateway reset.
RTP Multiplexing Remote UDP Port **[L1L1ComplexRxUDPPort]**	Determines the remote UDP port the multiplexed RTP packets are sent to, and the local UDP port used for incoming multiplexed RTP packets (applies to the ThroughPacket™ mechanism). The valid range is the range of possible UDP ports: 6,000 to 64,000. The default value is 0 (ThroughPacket™ is disabled). This parameter cannot be changed on-the-fly and requires a gateway reset. **Note:** All gateways that participate in the same ThroughPacket™ session must use the same 'L1L1ComplexRxUDPPort'.
RTCP XR Settings (For a detailed description of RTCP-XR reports, refer to Appendix F on page 427.)	
Enable RTCP XR **[VQMonEnable]**	Enables voice quality monitoring and RTCP Extended Reports (RTCP-XR). Valid options include: 0 = Disable (default) 1 = Enable

Table 5-30: Media Settings, RTP / RTCP Parameters (continues on pages 155 to 156)

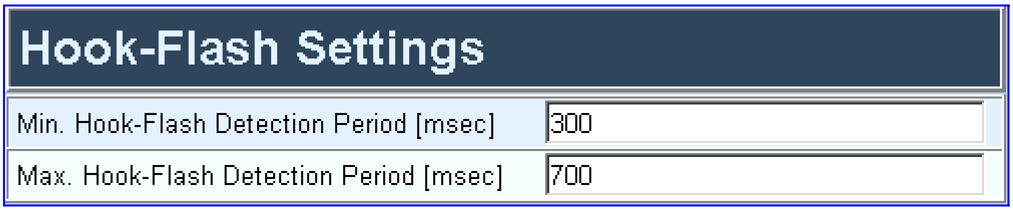
Parameter	Description
RTCP XR Report Mode **[RTCPXRReportMode]**	Determines whether or not RTCP-XR reports are sent to the Event State Compositor (ESC) and if so, defines the interval in which they are sent. Valid options include: 0 = Disable (RTCP-XR reports are not sent to the ESC) -- default 1 = End Call (RTCP-XR reports are sent to the ESC at the end of each call) 2 = End Call & Periodic (RTCP-XR reports are sent to the ESC at the end of each call and periodically according to the parameter RTCPInterval)
RTCP XR Packet Interval **[RTCPInterval]**	Defines the time interval (in msec) between adjacent RTCP reports. The interval range is 0 to 65,535. The default interval is 5,000.
Disable RTCP XR Interval Randomization **[DisableRTCPRandomize]**	Controls whether RTCP report intervals are randomized or whether each report interval accords exactly to the parameter RTCPInterval. Valid options include: 0 = Randomize (default) 1 = No Randomize
RTCP XR Collection Server **[RTCPXREscIP]**	IP address of the Event State Compositor (ESC). The gateway sends RTCP-XR reports using PUBLISH messages to this server. The address can be configured as a numerical IP address or as a domain name.

5.6.2.4 Configuring the Hook-Flash Settings

➢ **To configure the Hook-Flash Settings parameters, take these 4 steps:**

1. Open the ‘Hook-Flash Settings’ screen (**Advanced Configuration** menu > **Media Settings** > **Hook-Flash Settings** option); the ‘Hook-Flash Settings’ screen is displayed.

Figure 5-44: Hook-Flash Settings Screen



Hook-Flash Settings

Min. Hook-Flash Detection Period [msec]: 300

Max. Hook-Flash Detection Period [msec]: 700

2. Configure the Hook-Flash Settings according to Table 5-31.
3. Click the **Submit** button to save your changes.
4. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Table 5-31: Media Settings, Hook-Flash Settings Parameters

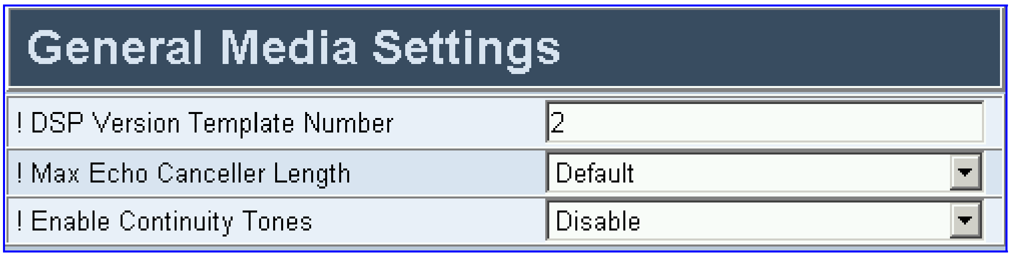
Parameter	Description
Min. Flash-Hook Detection Period [msec] **[MinFlashHookTime]**	Sets the minimal time (in msec) for detection of a flash-hook event (for FXS only). The valid range is 25 to 300. The default value is 300 msec. Detection is guaranteed for flash hook periods of at least 60 msec (when setting the minimal time to 25). Flash-hook signals that last a shorter period of time are ignored. **Note:** It's recommended to reduce the detection time by 50 msec from the desired value (e.g. if you set the value to 200 msec, then enter 150 msec (i.e. 200 minus 50).
Max. Flash-Hook Detection Period [msec] **[FlashHookPeriod]**	Defines the flash-hook period (in msec) for both analog and IP sides. For the analog side it defines the following: ▪ Maximal hook-flash detection period (for FXS gateways). A longer signal is considered offhook / onhook event. ▪ Hook-flash generation period (for FXO gateways). For the IP side it defines the flash-hook period that is reported to IP. The valid range is 25 to 1500. The default value is 700 msec. **Note:** For FXO gateways, a constant of 90 msec must be added to the required hook-flash period. For example, to generate a 450 msec hook-flash, set 'FlashHookPeriod' to 540.

5.6.2.5 Configuring the General Media Settings

➢ To configure the General Media Settings parameters, take these 4 steps:

1. Open the 'General Media Settings' screen (**Advanced Configuration** menu > **Media Settings** > **General Media Settings** option); the 'General Media Settings' screen is displayed.

Figure 5-45: General Media Settings Screen



General Media Settings

! DSP Version Template Number: 2

! Max Echo Canceller Length: Default

! Enable Continuity Tones: Disable

2. Configure the General Media Settings according to Table 5-32.
3. Click the **Submit** button to save your changes.
4. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Table 5-32: Media Settings, General Media Settings Parameters

Parameter	Description
DSP Version Template Number	N/A.
Max Echo Canceller Length [**MaxEchoCancellerLength**]	Maximum Echo Canceler Length in msec: 0 = based on various internal gateway settings -- 64 msec (default) 4 = 32 msec 11 = 64 msec **Note 1:** The gateway must be reset after the value of ‘MaxEchoCancellerLength’ is changed. **Note 2:** It isn’t necessary to configure the parameter EchoCancellerLength as it automatically acquires its value from the parameter MaxEchoCancellerLength.
Enable Continuity Tones	N/A.

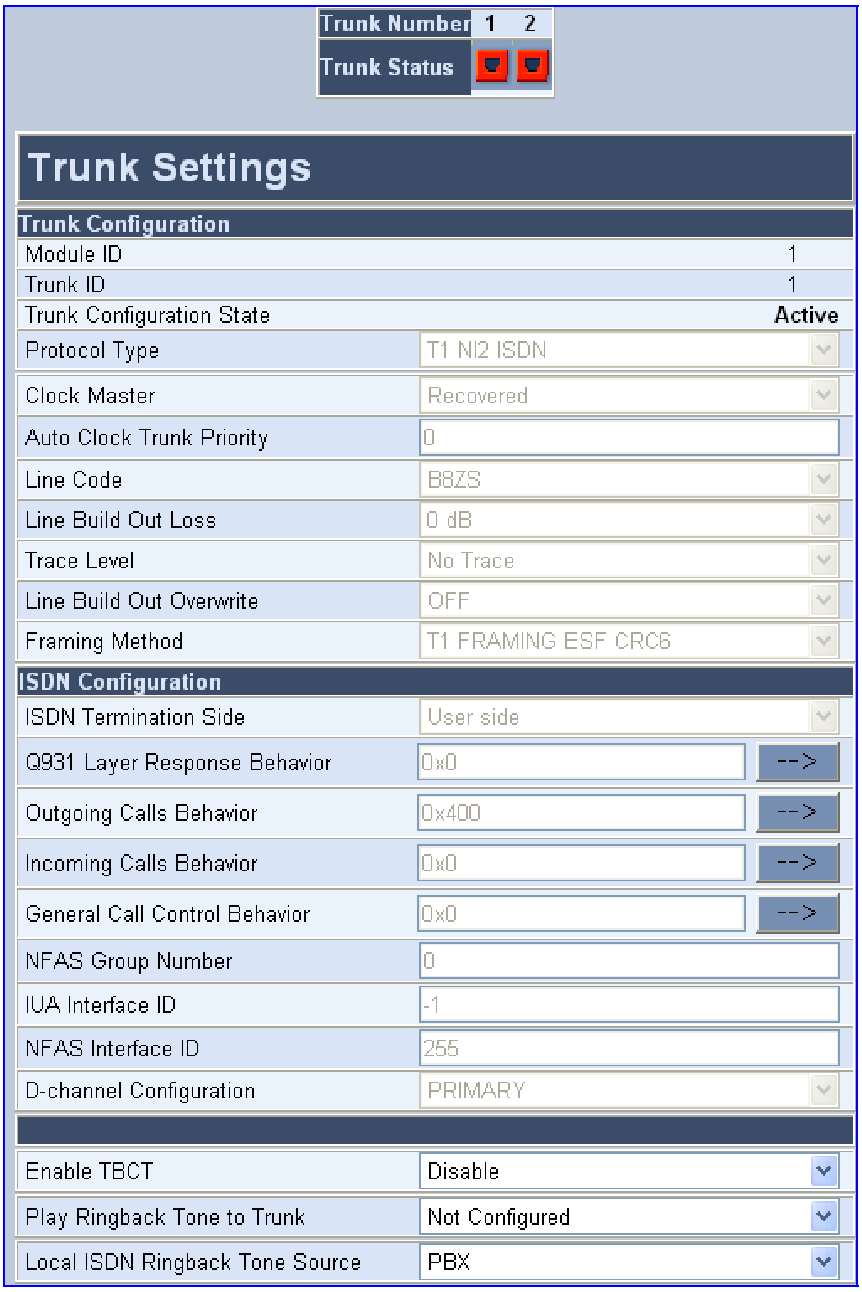
5.6.3 Configuring the Trunk Settings

➢ To configure the Trunk Settings, take these 9 steps:

1. Open the ‘Trunk Settings’ screen (**Advanced Configuration** menu > **Trunk Settings**); the ‘Trunk Settings’ screen is displayed.

   Initially, the screen appears with the parameters fields grayed (i.e., read-only). The **Stop Trunk** button appears at the bottom of the screen. The Trunk Status indicators appear colored. Table 5-46 on page 204 shows the possible indicators and their descriptions.

Figure 5-46: E1/T1 Trunk Settings Screen



Trunk Number	1	2
Trunk Status	🟥	🟥

Trunk Settings

Trunk Configuration	
Module ID	1
Trunk ID	1
Trunk Configuration State	Active
Protocol Type	T1 NI2 ISDN
Clock Master	Recovered
Auto Clock Trunk Priority	0
Line Code	B8ZS
Line Build Out Loss	0 dB
Trace Level	No Trace
Line Build Out Overwrite	OFF
Framing Method	T1 FRAMING ESF CRC6
ISDN Configuration	
ISDN Termination Side	User side
Q931 Layer Response Behavior	0x0
Outgoing Calls Behavior	0x400
Incoming Calls Behavior	0x0
General Call Control Behavior	0x0
NFAS Group Number	0
IUA Interface ID	-1
NFAS Interface ID	255
D-channel Configuration	PRIMARY
Enable TBCT	Disable
Play Ringback Tone to Trunk	Not Configured
Local ISDN Ringback Tone Source	PBX

2. To configure the parameters of a specific trunk from the trunks displayed on the top, select the trunk you want to configure by clicking the Trunk's Status indicator. The parameter 'Trunk ID' changes to reflect the trunk you clicked. The parameters displayed are for the selected trunk only.
3. To modify the selected trunk's parameters, click the **Stop Trunk** button; the trunk is deactivated, the status of the parameter 'Trunk Configuration State' changes to 'Inactive', the parameters are no longer grayed and can be modified.

> **Note:** If the trunk can't be stopped because it provides the gateway's clock (assuming the Mediant 1000 is synchronized with the E1/T1 clock), assign a different E1/T1 trunk to provide the gateway's clock or enable 'TDM Bus PSTN Auto Clock' on the 'TDM Bus Settings' screen (refer to Section 5.6.4 on page 166).
> To assign a different E1/T1 trunk that provides the gateway's clock, access the 'TDM Bus Setting' screen and change the 'TDM Bus Local Reference' number to any other trunk number (this operation can be performed on-the-fly).

4. Select the 'Protocol Type' you use. Note that different trunks can be defined with different protocols (CAS or ISDN variants) on the same gateway (subject to the constraints in the Mediant 1000 SIP Release Notes).

> **Note:** When modifying the 'Protocol Type' field, the menu is automatically updated according to the selected protocol (ISDN, CAS or Transparent). Additional parameters are appropriate to the selected protocol type.

5. Modify the relevant trunk configuration parameters according to your requirements.
6. To configure the different behavior bits, either enter the exact hexadecimal value of the bits in the field to the right of the relevant behavior parameter, or directly configure each bit field by completing the following steps:
   a. Click the arrow button (-->) to the right of the relevant behavior parameter; a new window appears.
   b. Modify each bit field according to your requirements.
   c. Click the **Submit** button to save your changes.
7. After modifying the parameters:
   - To apply the changes to the selected trunk only, click the **Apply Trunk Settings** button.
   - To apply the changes to all the trunks, click the **Apply to all Trunks** button.

   The screen is refreshed; parameters become read-only (indicated by being grayed); the **Stop Trunk** button appears at the bottom of the screen.
8. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

> **Note:** Some parameter configuration options require a device reset; when this is the case, the Web interface prompts the user.

9. To reset the Mediant 1000, refer to Section 5.9.3 on page 200.

Table 5-33: E1/T1/J1 Configuration Parameters (continues on pages 162 to 165)

ini File Field Name Web Parameter Name	Valid Range and Description
ProtocolType [Protocol Type]	Sets the PSTN protocol to be used for this trunk. 1 = E1_EURO_ISDN 2 = T1_CAS 3 = T1_RAW_CAS 4 = T1_TRANSPARENT 5 = E1_TRANSPARENT_31 6 = E1_TRANSPARENT_30 7 = E1_MFCR2 8 = E1_CAS_R2 9 = E1_RAW_CAS 10 = T1_NI2_ISDN 11 = T1_4ESS_ISDN 12 = T1_5ESS_9_ISDN 13 = T1_5ESS_10_ISDN 14 = T1_DMS100_ISDN 15 = J1_TRANSPARENT 16 = T1_NTT_ISDN (Japan - Nippon Telegraph) 17 = E1_AUSTEL_ISDN (Australian Telecom) 18 = T1_HKT_ISDN (Hong Kong – HKT) 19 = E1_KOR_ISDN (Korean operator) 20 = T1_HKT_ISDN (Hong Kong - HKT over T1) 21 = E1_QSIG 23 = T1_QSIG 31 = E1_FRENCH_VN3_ISDN 35 = T1_DMS100_Meridian 40 = E1_NI2_ISDN **Note:** The gateway simultaneously supports different variants of CAS and PRI protocols on different E1/T1 spans (no more than four simultaneous PRI variants).
ClockMaster [Clock Master]	Determines the Tx clock source of the E1/T1 line. 0 = Generate the clock according to the Rx of the E1/T1 line (default). 1 = Generate the clock according to the internal TDM bus. For detailed information on configuring the gateway's clock settings, refer to Section 11.1 on page 315. **Note:** The source of the internal TDM bus clock is determined by the parameter TDMBusClockSource.
AutoClockTrunkPriority [Auto Clock Trunk Priority]	Defines the trunk priority for auto-clock fallback (per trunk parameter). 0 to 99 = priority (0 is the highest = default). 100 = the SW never performs a fallback to that trunk (usually used to mark un-trusted source of clock). **Note:** Fallback is enabled when the TDMBusPSTNAutoClockEnable parameter is set to 1.
LineCode [Line Code]	0 = use B8ZS line code (for T1 trunks only) default. 1 = use AMI line code. 2 = use HDB3 line code (for E1 trunks only). Use to select B8ZS or AMI for T1 spans, and HDB3 or AMI for E1 spans.
LineBuildOut.Loss [Line Build Out Loss]	Selects the line build out loss to be used for T1 trunks. 0 = 0 dB (default) 1 = -7.5 dB 2 = -15 dB 3 = -22.5 dB **Note:** Not applicable for E1 trunks.

Table 5-33: E1/T1/J1 Configuration Parameters (continues on pages 162 to 165)

ini File Field Name Web Parameter Name	Valid Range and Description
TraceLevel [Trace Level]	Defines the trace level. Valid options include: 0 = No trace (default) 1= Full ISDN trace 2 = Layer 3 ISDN trace 3 = Only ISDN Q.931 messages trace 4 = Layer 3 ISDN no duplication trace
FramingMethod [Framing Method]	Selects the framing method to be used for E1/T1 spans. **For E1** 0 = Multiframe with CRC4 (default, automatic mode, if CRC is identified in the Rx, CRC is sent in Tx, otherwise no CRC). a = Double frame c = Multiframe with CRC4 **For T1** 0 or D = Extended super frame with CRC6 (default) 1 or B = Super frame D4, F12 (12-Frame multiframe) A = F4 (4-Frame multiframe) C = Extended super frame without CRC6 F = J1 - Japan (ESF with CRC6 and JT)
ISDN Configuratuion Parameters	
TerminationSide [ISDN Termination Side]	Selects the ISDN termination side. Applicable only to ISDN protocols. 0 = ISDN User Termination Side (TE) (default) 1 = ISDN Network Termination Side (NT) **Note:** select 'User Side' when the PSTN or PBX side is configured as 'Network side', and vice-versa. If you don't know the gateway ISDN termination side, choose 'User Side' and refer to the 'Status & Diagnostics>Channel Status' screen. If the D-channel alarm is indicated, choose 'Network Side'.
NFASGroupNumber_x [NFAS Group Number]	0 = Non NFAS trunk (default) 1 to 4 = NFAS group number Indicates the NFAS group number (NFAS member) for the selected trunk. 'x' identifies the Trunk ID. Trunks that belong to the same NFAS group have the same number. With ISDN Non-Facility Associated Signaling you can use single D-channel to control multiple PRI interfaces. Applicable only to T1 ISDN protocols.
ISDNNFASInterfaceID_x [NFAS Interface ID]	Defines a different Interface ID for each T1 trunk. The valid range is 0 to 100. The default interface ID equals to the trunk's ID. 'x' identifies the trunk ID. **Note:** To set the NFAS interface ID, configure: ISDNIBehavior_x to include '512' feature, per T1 trunk.
DChConfig_x [D-channel Configuration]	0 = Primary Trunk (default) 1 = Backup Trunk 2 = NFAS Trunk D-channel configuration parameter defines primary, backup (optional) and B-channels only trunks. 'x' identifies the Trunk ID. Primary trunk contains D-channel that is used for signaling. Backup trunk contains backup D-channel that is used if the primary D-channel fails. The other NFAS trunks contain only 24 B-channels, without a signaling D-channel. **Note:** Applicable only to T1 ISDN protocols.

Table 5-33: E1/T1/J1 Configuration Parameters (continues on pages 162 to 165)

***ini* File Field Name** **Web Parameter Name**	**Valid Range and Description**
ISDNRxOverlap_x [Enable Receiving of Overlap Dialing]	Enable / disable Rx ISDN overlap per trunk ID. 0 = Disabled (default). 1 = Enabled. **Note 1:** If enabled, the gateway receives ISDN called number that is sent in the 'Overlap' mode. **Note 2:** The SETUP message to IP is sent only after the number (including the 'Sending Complete' Info Element) was fully received (via SETUP and/or subsequent INFO Q.931 messages). **Note3:** The 'MaxDigits' parameter can be used to limit the length of the collected number for gateway ISDN overlap dialing (if sending complete was not received). **Note 4:** If a digit map pattern is defined (DigitMapping), the gateway collects digits until a match is found (e.g., useful for closed numbering schemes) or until a timer expires (e.g., useful for open numbering schemes). If a match is found (or the timer expires), the digit collection process is terminated even if Sending Complete wasn't received.
HeldTimeout	For a description of this parameter, refer to Section 6.5.12 page 253.
ISDN Flexible Behavior Parameters ISDN protocol is implemented in different Switches / PBXs by different vendors. Several implementations vary a little from the specification. Therefore, to provide a flexible interface that supports these ISDN variants, the ISDN behavior parameters are used.	
ISDNIBehavior [Q.931 Layer Response Behavior]	▪ 1 = Q.931 Status message isn't sent if Q.931 received message contains an unknown/unrecognized IE(s). By default the Status message is sent. This parameter applies only to PRI variants in which sending of Status message is optional. ▪ 2 = Q.931 Status message isn't sent if an optional IE with invalid content is received. By default the Status message is sent. This parameter applies only to PRI variants in which sending of Status message is optional. ▪ 4 = Accepts unknown/unrecognized Facility IE. Otherwise, (default) the Q.931 message that contains the unknown Facility IE is rejected. This parameter applies to PRI variants where a complete ASN1 decoding is performed on Facility IE. ▪ 128 = Connect ACK message is sent in response to received Q.931 Connect. Applicable only to Euro ISDN User side outgoing calls. Otherwise, the Connect ACK is not sent (default). ▪ 512 = Enables to configure T1 NFAS Interface ID (refer to the parameter 'ISDNNFASInterfaceID_x'). Applicable to 4/5ESS, DMS, NI-2 and HKT variants. ▪ 2048 = Always set the Channel Identification IE to explicit Interface ID, even if the B-channel is on the same trunk as the D-channel. Applicable to 4/5ESS, DMS and NI-2 variants. ▪ 65536 = Applicable to ETSI, NI-2 and 5ESS. The calling party number (octet 3a) is always present even when presentation and screening are at their default. ▪ 131072 = Clears the call on reception of Q.931 Status with incompatible state. Otherwise, (default) no action is taken. **Note:** To configure the gateway to support several 'ISDNIBehavior' features, summarize the individual feature values. For example to support both '512' and '2048' features, set 'ISDNIBehavior = 2560'.
ISDNOutCallsBehavior [Outgoing Calls Behavior]	1024 = Numbering plan / type for T1 IP→Tel calling number are defined according to the manipulation tables or according to RPID header (default). Otherwise, the Plan / type for T1 calls are set according to the length of the calling number

Table 5-33: E1/T1/J1 Configuration Parameters (continues on pages 162 to 165)

***ini* File Field Name** **Web Parameter Name**	**Valid Range and Description**
ISDNInCallsBehavior [Incoming Calls Behavior]	2048 = Sends Channel ID in the first response to an incoming Q.931 Call Setup message. Otherwise, the Channel ID is sent only if the gateway requires changing the proposed Channel ID (default). 8192 = Sends Channel ID in a Q.931 Call Proceeding message. 65536 = Includes Progress Indicator (PI=8) in Setup ACK message, if an empty called number is received in an incoming Setup message. Applicable to overlap dialing mode. The parameter also directs the gateway to play a dial tone (for 'TimeForDialTone'), until the next called number digits are received. 262144 = NI-2 second redirect number – Users can select and use (in INVITE messages) the NI-2 second redirect number, if two redirect numbers are received in Q.931 Setup for incoming Tel→IP calls. **Note:** To configure the gateway to support several 'ISDNInCallsBehavior' features, summarize the individual feature values. For example to support both '2048' and '65536' features, set 'ISDNInCallsBehavior = 67584.
ISDNGeneralCCBehavior [General Call Control Behavior]	16 = The gateway clears down the call if it receives a NOTIFY message specifying 'User-Suspended'. A NOTIFY (User-Suspended) message is used by some networks (e.g., in Italy or Denmark) to indicate that the remote user has cleared the call, especially in the case of a long distance voice call. 32 = Applies only to ETSI E1 lines (30B+D). Enables handling the differences between the newer QSIG standard (ETS 300-172) and other ETSI-based standards (ETS 300-102 and ETS 300-403) in the conversion of B-channel ID values into timeslot values: 1) In 'regular ETSI' standards, the timeslot is identical to the B-channel ID value, and the range for both is 1 to 15 and 17 to 31. The D-channel is identified as channel-id #16 and carried into the timeslot #16. 2) In newer QSIG standards, the channel-id range is 1 to 30, but the timeslot range is still 1 to 15 and 17 to 31. The D-channel is not identified as channel-id #16, but is still carried into the timeslot #16. When this bit is set, the channel ID #16 is considered as a valid B-channel ID, but timeslot values are converted to reflect the range 1 to 15 and 17 to 31. This is the new QSIG mode of operation. When this bit is not set (default), the channel_id #16 is not allowed, as for all ETSI-like standards.
CAS Configuration	

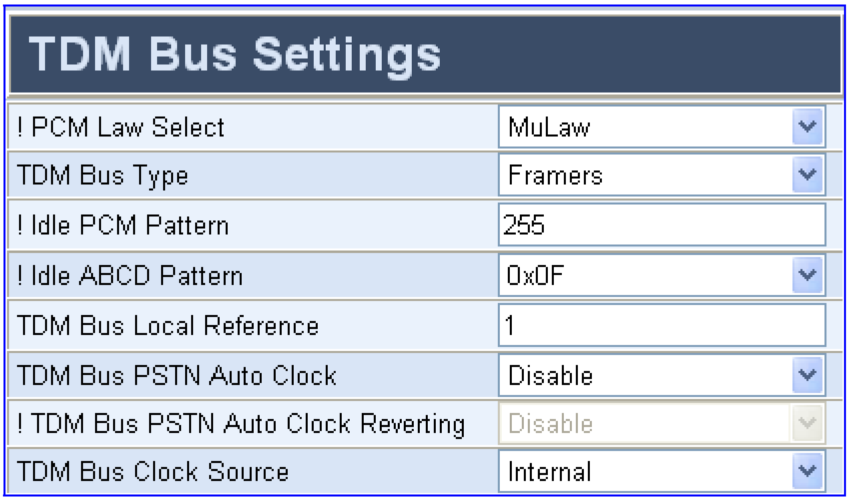
CASTableIndex_x
[CAS Table]
	Defines CAS protocol for each trunk ID from a list of CAS protocols defined by the parameter CASFileName_Y.
	For example:
	CASFileName_0 = 'E_M_WinkTable.dat'
	CASFileName_1 = 'E_M_ImmediateTable.dat'
	CASTableIndex_0 = 0
	CASTableIndex_1 = 0
	CASTableIndex_2 = 1
	CASTableIndex_3 = 1
	Trunks 0 and 1 use the E&M Winkstart CAS protocol, while trunks 2 and 3 use the E&M Immediate Start CAS protocol.
	Note: For additional CAS table *ini* file parameters (CASFileName_0, CASFileName_1, CASFileName_7, and CASTablesNum), refer to Section 6.5.12 on page 253.

5.6.4 Configuring the TDM Bus Settings

➢ To configure the TDM Bus Settings parameters, take these 5 steps:

1. Open the 'TDM Bus Settings' screen (**Advanced Configuration** menu > **TDM Bus Settings**); the 'TDM Bus Settings' screen is displayed.

Figure 5-47: TDM Bus Settings Screen



TDM Bus Settings

! PCM Law Select	MuLaw
TDM Bus Type	Framers
! Idle PCM Pattern	255
! Idle ABCD Pattern	0x0F
TDM Bus Local Reference	1
TDM Bus PSTN Auto Clock	Disable
! TDM Bus PSTN Auto Clock Reverting	Disable
TDM Bus Clock Source	Internal

2. Configure the TDM Bus Settings parameters.
3. Click the **Submit** button to save your changes.
4. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.
5. A device reset is required to activate the TDM Bus Settings parameters. To reset the Mediant 1000, refer to Section 5.9.3 on page 200.

Note: Usually the 'PCM Law Select' parameter is set to A-law for E1 trunks and to µ-law for T1 trunks.

Refer to Section 11.1 on page 315 for information on configuring the 'TDM Bus Clock Source', 'TDM Bus Enable Fallback' and 'TDM Bus PSTN Auto Clock' parameters.

Table 5-34: TDM Bus Settings Parameters (continues on pages 166 to 167)

Parameter	Description
PCM Law Select **[PCMLawSelect]**	1 = Alaw (default) 3 = MuLaw Usually A-Law is used for E1 spans and µ-Law for T1 and J1 spans.
TDM Bus Type **[TDMBusType]**	N/A.
Idle PCM Pattern **[IdlePCMPattern]**	PCM Pattern to be applied to E1/T1 timeslot (B-channel) when the channel is idle. Range 0x00 to 0xFF. Default = -1 (default pattern = 0xFF for µ-Law, 0x55 for A-law).

Table 5-34: TDM Bus Settings Parameters (continues on pages 166 to 167)

Parameter	Description
Idle ABCD Pattern **[IdleABCDPattern]**	ABCD (CAS) Pattern to be applied to CAS signaling bus when the channel is idle. Range 0x0 to 0xF Default = -1 (default, pattern = 0000) **Note:** This is only relevant when using PSTN interface with CAS protocols.
TDM Bus Local Reference **[TDMBusLocalReference]**	0 to 4 (default = 0) Physical Trunk ID from which the gateway recovers its clock. **Note:** Applicable only if TDMBusClockSource = 4 and TDMBusPSTNAutoClockEnable = 0.
TDM Bus PSTN Auto Clock **[TDMBusPSTNAutoClockEnable]**	0 = Recovers the clock from the E1/T1 line defined by the parameter TDMBusLocalReference (default) 1 = Recovers the clock from any connected synchronized slave E1/T1 line. If this trunk loses its synchronization, the gateway attempts to recover the clock from the next trunk. Note that initially the gateway attempts to recover the clock from the trunk defined by the parameter TDMBusLocalReference. **Note:** This parameter is relevant only if TDMBusClockSource = 4.
TDM Bus PSTN Auto Clock Reverting **[TDMBusPSTNAutoClockRevertingEnable]**	Enables or disables the PSTN trunk auto-fallback reverting feature. If a trunk with a higher priority than the current LocalReference is being synchronized, the board LocalReference changes to the new trunk. Valid options include: 0 = Disable (default) 1 = Enable **Note:** The parameter is valid only when the TDMBusPSTNAutoClockEnable parameter is set to 1.
TDM Bus Clock Source **[TDMBusClockSource]**	1 = Generate clock from local source (default). 4 = Recover clock from PSTN line. For detailed information on configuring the gateway's clock settings, refer to Section 11.1 on page 315.

5.6.5 Restoring and Backing up the Gateway Configuration

The Configuration File screen enables you to restore (load a new *ini* file to the gateway) or to back up (make a copy of the VoIP gateway *ini* file and store it in a directory on your computer) the current configuration the gateway is using.

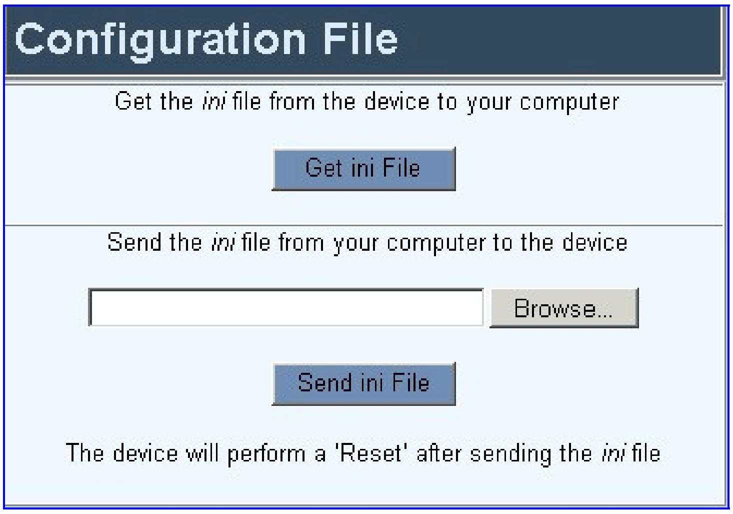
Back up your configuration if you want to protect your VoIP gateway programming. The backup *ini* file includes only those parameters that were modified and contain other than default values.

Restore your configuration if the VoIP gateway has been replaced or has lost its programming information, you can restore the VoIP gateway configuration from a previous backup or from a newly created *ini* file. To restore the VoIP gateway configuration from a previous backup you must have a backup of the VoIP gateway information stored on your computer.

➢ To restore or back up the *ini* file, take this step:

- Open the 'Configuration File' screen (**Advanced Configuration** menu > **Configuration File**); the 'Configuration File' screen is displayed.

Figure 5-48: Configuration File Screen



Configuration File
Get the *ini* file from the device to your computer
Get ini File
Send the *ini* file from your computer to the device
Browse...
Send ini File
The device will perform a 'Reset' after sending the *ini* file

➢ To back up the *ini* file on your PC, take these 4 steps:

1. Click the **Get ini File** button; the 'File Download' window opens.
2. Click the **Save** button; the 'Save As' window opens.
3. Navigate to the folder where you want to save the *ini* file on your PC.
4. Click the **Save** button; the VoIP gateway copies the *ini* file into the folder you selected.

➢ To restore the *ini* file, take these 4 steps:

1. Click the **Browse** button.
2. Navigate to the folder that contains the *ini* file you want to load.
3. Click the file and click the **Open** button; the name and path of the file appear in the field beside the Browse button.
4. Click the **Send *ini* File** button, and click **OK** in the prompt; the gateway is automatically reset (from the *cmp* version stored on the flash memory).

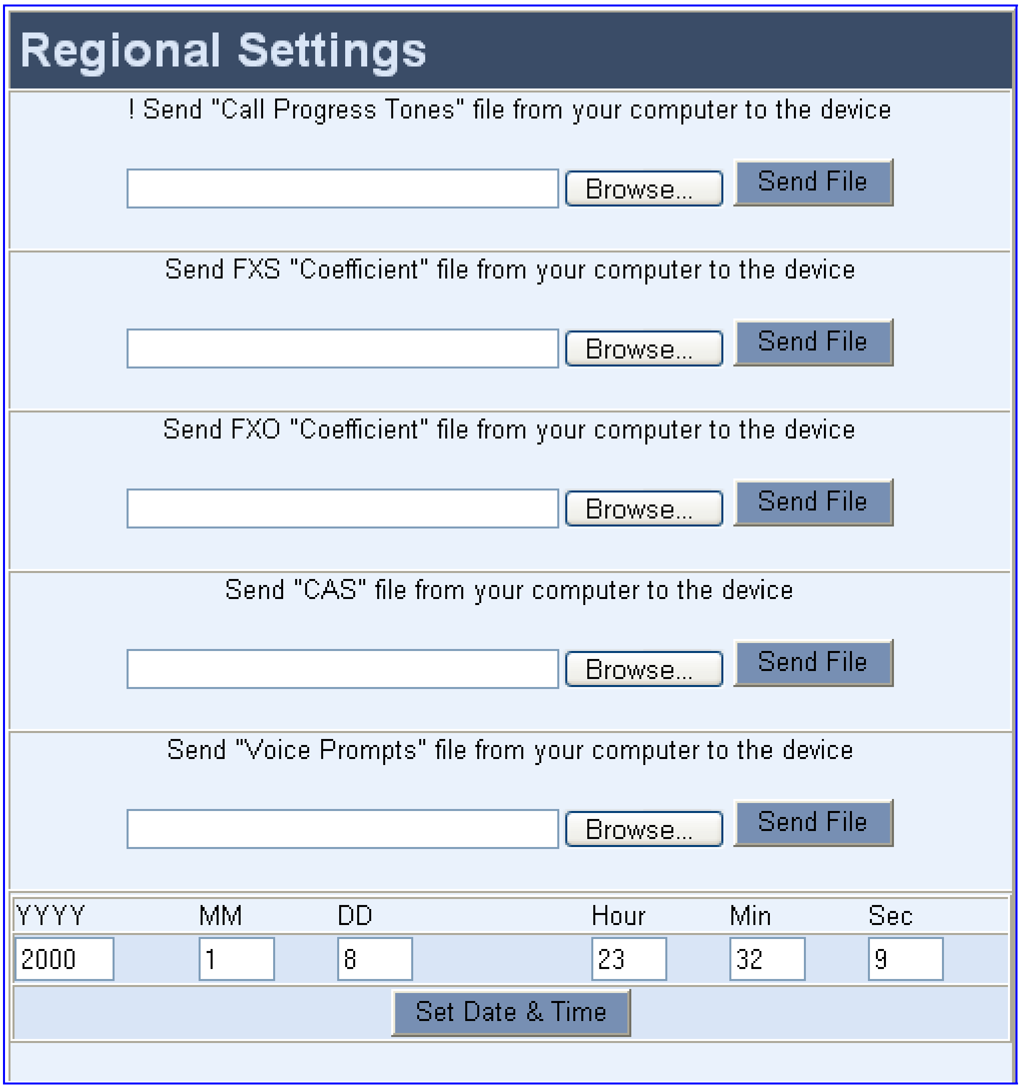
5.6.6 Regional Settings

The ‘Regional Settings’ screen allows you to define and view the gateway’s internal date and time as well as to load configuration files (Call Progress Tones, FXS/FXO coefficient, and CAS) to the gateway. For detailed information on the configuration files, refer to Section 5.8.2 on page 196.

➢ To configure the date and time of the Mediant 1000, take these 3 steps:

1. Open the ‘Regional Settings’ screen (**Advanced Configuration** menu > **Regional Settings**); the ‘Regional Settings' screen is displayed.

Figure 5-49: Regional Settings Screen



Regional Settings

! Send "Call Progress Tones" file from your computer to the device

Send FXS "Coefficient" file from your computer to the device

Send FXO "Coefficient" file from your computer to the device

Send "CAS" file from your computer to the device

Send "Voice Prompts" file from your computer to the device

YYYY	MM	DD	Hour	Min	Sec
2000	1	8	23	32	9

2. Enter the time and date where the gateway is installed.
3. Click the **Set Date & Time** button; the date and time are automatically updated.

Note: After performing a hardware reset, the date and time are returned to their defaults and should therefore, be updated.

➢ To load a configuration file to the VoIP gateway, take these 8 steps:

1. Open the ‘Regional Settings’ screen (**Advanced Configuration** menu > **Regional Settings**); the ‘Regional Settings’ screen is displayed (shown in Figure 5-49).
2. Click the **Browse** button adjacent to the file you want to load.
3. Navigate to the folder that contains the file you want to load.
4. Select the file, and then click the **Open** button; the name and path of the file appear in the field beside the **Browse** button.
5. Click the **Send File** button that is next to the field that contains the name of the file you want to load. An exclamation mark in the screen section indicates that the file’s loading doesn’t take effect on-the-fly (e.g., CPT file).
6. Repeat steps 2 to 5 for each file you want to load.
7. To save the loaded auxiliary files so they are available after a power fail, refer to Section 5.9.2 on page 200. (Refer to note below.)
8. To reset the Mediant 1000, refer to Section 5.9.3 on page 200.

⚠

Notes:

- Saving a configuration file to flash memory may disrupt traffic on the Mediant 1000. To avoid this, disable all traffic on the gateway before saving to flash memory by performing a graceful lock (refer to Section 5.9.1 on page 198).
- A device reset is required to activate a loaded CPT file.

5.6.7 Security Settings

From the Security Settings you can:

- Configure the Web User Accounts (refer to Section 5.6.7.1 below)
- Configure the Web & Telnet Access List (refer to Section 5.6.7.2 on page 172)
- Configure the Firewall Settings (refer to Section 5.6.7.3 on page 172)
- Configure the Certificates (refer to Section 5.6.7.4 on page 174)
- Configure the General Security Settings (refer to Section 5.6.7.5 on page 175)
- Configure the IPSec Table (refer to Section 5.6.7.6 on page 177)
- Configure the IKE Table (refer to Section 5.6.7.7 on page 178)

5.6.7.1 Configuring the Web User Accounts

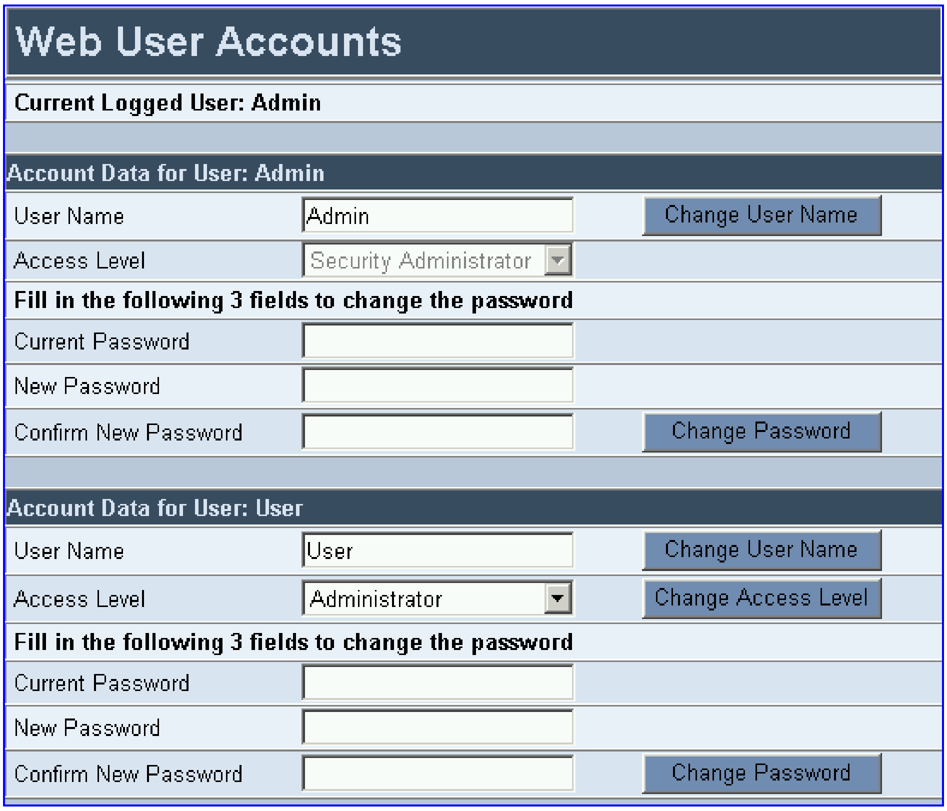
To prevent unauthorized access to the Embedded Web Server, two user accounts are available, a primary and secondary. Each account is composed of three attributes: username, password and access level. For detailed information on the user account mechanism, refer to Section 5.2.1 on page 55.

It is recommended that you change the default username and password of the account you use to access the Embedded Web Server.

➢ To change the Web User Accounts attributes, take these 4 steps:

1. Open the ‘Web User Accounts’ screen (**Advanced Configuration** menu > **Security Settings** > **Web User Accounts** option); the ‘Web User Accounts’ screen is displayed.

Figure 5-50: Web User Accounts Screen (for Users with ‘Security Administrator’ Privileges)



Web User Accounts

Current Logged User: Admin

Account Data for User: Admin

User Name	Admin	Change User Name
Access Level	Security Administrator ▾	
Fill in the following 3 fields to change the password		
Current Password		
New Password		
Confirm New Password		Change Password

Account Data for User: User

User Name	User	Change User Name
Access Level	Administrator ▾	Change Access Level
Fill in the following 3 fields to change the password		
Current Password		
New Password		
Confirm New Password		Change Password

2. To change the access level of the secondary account (the access level of the primary account cannot be changed), from the ‘Access Level’ drop-down list, select the new access level, and then click the button **Change Access Level**; the new access level is applied immediately.

3. To change the username of an account, enter the new username in the field ‘User Name’, and then click the button **Change User Name**; the new username is applied immediately and the ‘Enter Network Password’ screen appears (shown in Figure 5-1 on page 57). Enter the updated username in the ‘Enter Network Password’ screen. Note that the username can be a maximum of 19 case-sensitive characters.

4. To change the password of an account, enter the current password in the field ‘Current Password’, the new password in the fields ‘New Password’ and ‘Confirm New Password’, and then click the button **Change Password**; the new password is applied immediately and the ‘Enter Network Password’ screen appears (shown in Figure 5-1 on page 57). Enter the updated password in the ‘Enter Network Password’ screen. Note that the password can be a maximum of 19 case-sensitive characters.

Note: A user with a ‘Security Administrator’ access level can change all attributes for all accounts. Users with an access level other than ‘Security Administrator’ can only change their own password and username.

5.6.7.2 Configuring the Web and Telnet Access List

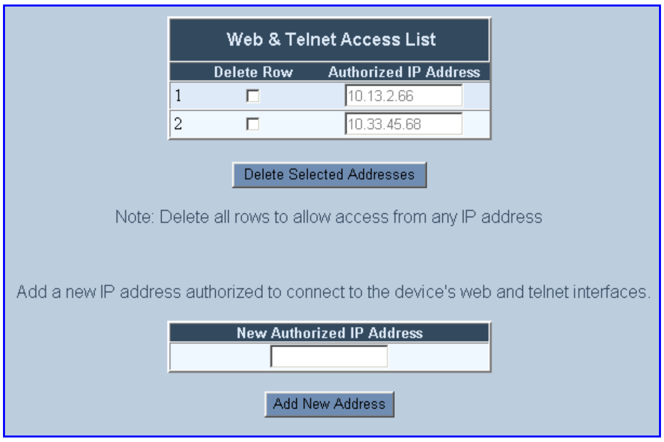
Use this screen to define up to ten IP addresses that are permitted to access the gateway's Web and Telnet interfaces. Access from an undefined IP address is denied. This security feature is inactive (the gateway can be accessed from any IP address) when the table is empty.

The Web and Telner Access List can also be defined using the *ini* file parameter WebAccessList_x (refer to Section 6.5.3 on page 225).

➢ To manage the Web & Telnet access list, take these 4 steps:

1. Open the 'Web & Telnet Access List' screen (**Advanced Configuration** menu > **Security Settings** > **Web & Telnet Access List** option); the 'Web & Telnet Access List' screen is displayed.

Figure 5-51: Web & Telnet Access List Screen



Web & Telnet Access List

	Delete Row	Authorized IP Address
1	□	10.13.2.66
2	□	10.33.45.68

Note: Delete all rows to allow access from any IP address

Add a new IP address authorized to connect to the device's web and telnet interfaces.

New Authorized IP Address

2. To add a new authorized IP address, in the 'New Authorized IP Address' field, enter the required IP address (refer to Note 1 below) and click the button **Add New Address**; the IP address you entered is added as a new entry to the Web & Telnet Access List table.
3. To delete authorized IP addresses, select the Delete Row checkbox in the rows of the IP addresses you want to delete (refer to Note 2 below) and then click the button **Delete Selected Addresses**; the IP addresses are removed from the table and can no longer access the Web and Telnet interfaces.
4. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Note 1: The first authorized IP address you add must be your own terminal's IP address. If it isn't, further access from your terminal is denied.

Note 2: Delete your terminal's IP address from the Web & Telnet Access List last. If it is deleted before the last, access from your terminal is denied from the point of its deletion onward.

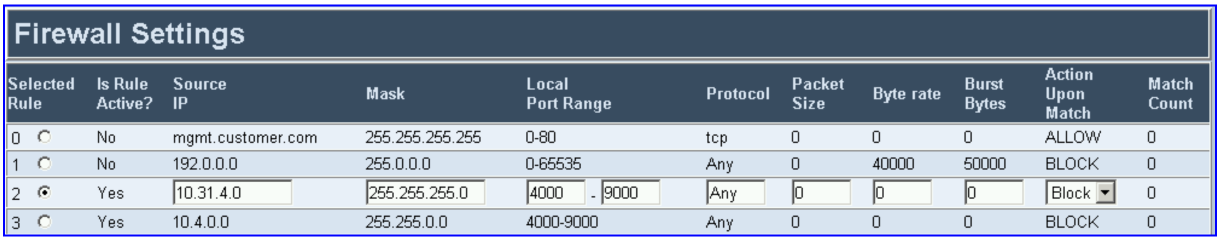
5.6.7.3 Configuring the Firewall Settings

The Mediant 1000 accommodates an internal Firewall, allowing the security administrator to define network traffic filtering rules. For detailed information on the internal Firewall, refer to Section 14.5 on page 360.

➢ To create a new access rule, take these 6 steps:

1. Open the ‘Firewall Settings’ screen (**Advanced Configuration** menu > **Security Settings** > **Firewall Settings** option); the ‘Firewall Settings’ screen is displayed.

Figure 5-52: Firewall Settings Screen



Firewall Settings

Selected Rule		Is Rule Active?	Source IP	Mask	Local Port Range	Protocol	Packet Size	Byte rate	Burst Bytes	Action Upon Match	Match Count
0	○	No	mgmt.customer.com	255.255.255.255	0-80	tcp	0	0	0	ALLOW	0
1	○	No	192.0.0.0	255.0.0.0	0-65535	Any	0	40000	50000	BLOCK	0
2	●	Yes	10.31.4.0	255.255.255.0	4000 - 9000	Any	0	0	0	Block	0
3	○	Yes	10.4.0.0	255.255.0.0	4000-9000	Any	0	0	0	BLOCK	0

2. In the ‘New Rule Index’ field, enter the index of the access rule that you want to add.
3. Click the **Add an Empty Rule** button; a new rule appears; alternatively, click the **Copy Selected Rule as a New Rule** button; a new rule that is an exact copy of the currently selected rule appears.
4. Configure the rule’s parameters according to Table 5-35.
5. Click one of the following buttons:
   - **Apply Rule Settings** to save the new rule (the rule isn’t active).
   - **Activate Rule** to save the new rule and activate it.
   - **Delete Rule** to delete the rule.
6. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

➢ To edit a rule, take these 5 steps:

1. Select the radio button of the entry you want to edit.
2. Click the **Make Rule Editable** button; the rule’s fields can now be modified.
3. Modify the fields according to your requirements.
4. Click the **Apply Rule Settings** button to save the changes.
5. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

➢ To activate a de-activated rule, take these 2 steps:

1. Select the radio button of the entry you want to activate.
2. Click the **Activate Rule** button; the rule is active.

➢ To de-activate an activated rule, take these 2 steps:

1. Select the radio button of the entry you want to activate.
2. Click the **DeActivate Rule** button; the rule is de-activated.

➢ To delete a rule, take these 3 steps:

1. Select the radio button of the entry you want to activate.
2. Click the **Delete Rule** button; the rule is deleted.
3. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Table 5-35: Internal Firewall Parameters (continues on pages 174 to 174)

Parameter	Description
Is Rule Active	A read-only field that indicates whether the rule is active or not. **Note:** After reset all rules are active.
Source IP **[AccessList_Source_IP]**	IP address (or DNS name) of source network, or a specific host.
Mask **[AccessList_Net_Mask]**	IP network mask. 255.255.255.255 for a single host or the appropriate value for the source IP addresses. The IP address of the sender of the incoming packet is bitwise ANDed with this mask and then compared to the field 'Source IP'.
Local Port Range **[AccessList_Start_Port]** **[AccessList_End_Port]**	The destination UDP/TCP ports (on this device) to which packets are sent. The valid range is 0 to 65535. **Note:** When the protocol type isn't TCP or UDP, the entire range must be provided.
Protocol **[AccessList_Protocol]**	The protocol type (e.g., UDP, TCP, ICMP, ESP or 'Any'), or the IANA protocol number (in the range of 0 (Any) to 255). **Note:** The protocol field also accepts the abbreviated strings 'SIP', 'MGCP', 'MEGACO' and 'HTTP'. Specifying these strings implies selection of the TCP or UDP protocols, and the appropriate port numbers as defined on the device.
Packet Size **[AccessList_Packet_Size]**	Maximum allowed packet size. The valid range is 0 to 65535. **Note:** When filtering fragmented IP packets, the Packet Size field relates to the overall (reassembled) packet size, not to the size of each fragment.
Byte Rate **[AccessList_Byte_Rate]**	Expected traffic rate (bytes per second).
Burst Bytes **[AccessList_Byte_Burst]**	Tolerance of traffic rate limit (number of bytes).
Action Upon Match **[AccessList_Allow_Type]**	Action upon match (allow or block).
Match Count **[AccessList_MatchCount]**	A read-only field that provides the number of packets accepted / rejected by a specific rule.

5.6.7.4 Configuring the Certificates

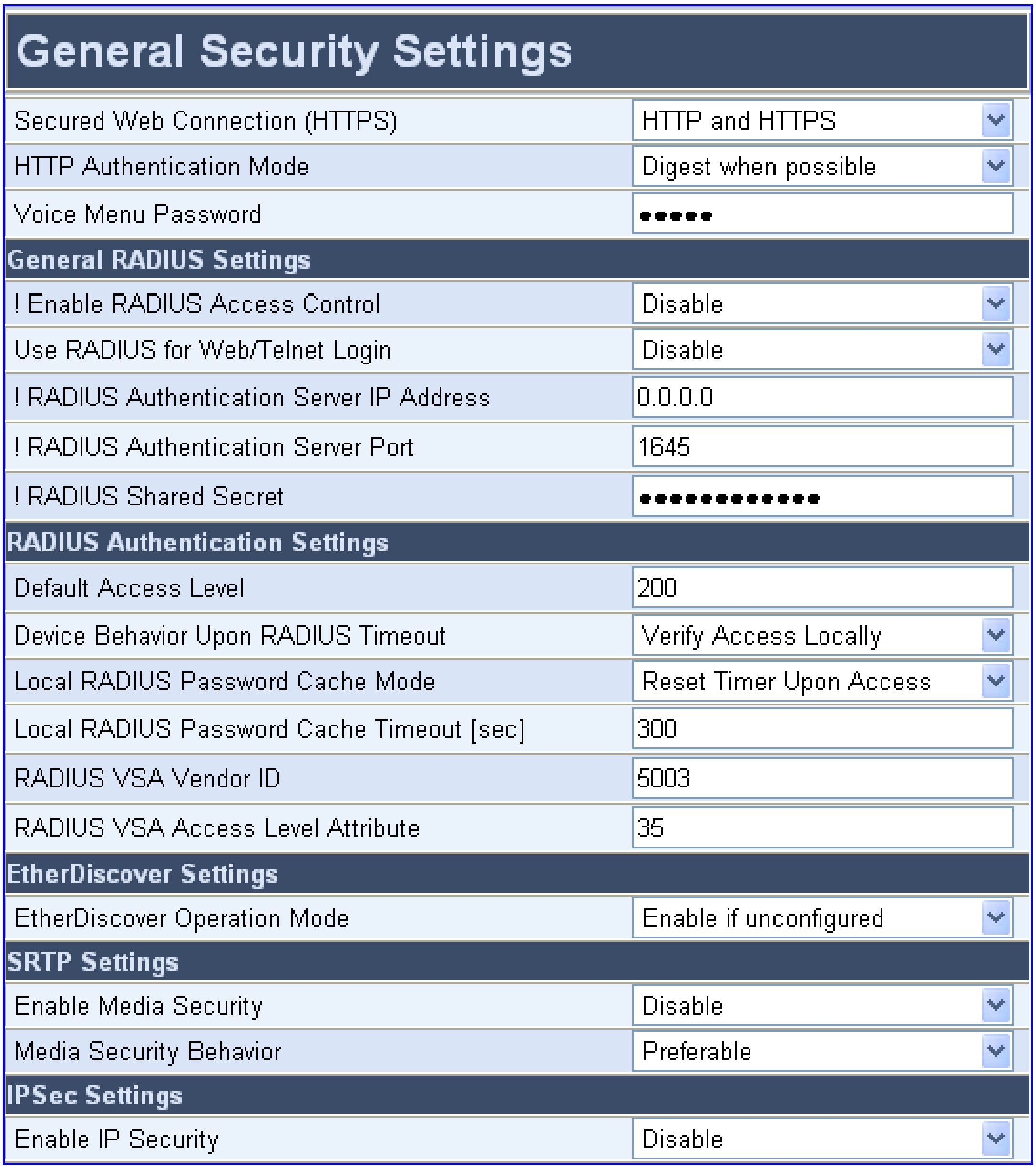
Use the Certificates screen to replace the server (refer to Section 14.2.4 on page 353) and client (refer to Section 14.2.5 on page 355) certificates and to update the private key (HTTPSPkeyFileName, described in Section 6.5.4 on page 227).

5.6.7.5 Configuring the General Security Settings

➢ **To configure the General Security Settings parameters, take these 4 steps:**

1. Open the 'General Security Settings' screen (**Advanced Configuration** menu > **Security Settings** > **General Security Settings** option); the 'General Security Settings' screen is displayed.

Figure 5-53: General Security Settings Screen



General Security Settings

Secured Web Connection (HTTPS)	HTTP and HTTPS
HTTP Authentication Mode	Digest when possible
Voice Menu Password	●●●●●
General RADIUS Settings	
! Enable RADIUS Access Control	Disable
Use RADIUS for Web/Telnet Login	Disable
! RADIUS Authentication Server IP Address	0.0.0.0
! RADIUS Authentication Server Port	1645
! RADIUS Shared Secret	●●●●●●●●●●●●●
RADIUS Authentication Settings	
Default Access Level	200
Device Behavior Upon RADIUS Timeout	Verify Access Locally
Local RADIUS Password Cache Mode	Reset Timer Upon Access
Local RADIUS Password Cache Timeout [sec]	300
RADIUS VSA Vendor ID	5003
RADIUS VSA Access Level Attribute	35
EtherDiscover Settings	
EtherDiscover Operation Mode	Enable if unconfigured
SRTP Settings	
Enable Media Security	Disable
Media Security Behavior	Preferable
IPSec Settings	
Enable IP Security	Disable

2. Configure the General Security Settings according to Table 5-36.
3. Click the **Submit** button to save your changes.
4. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Table 5-36: General Security Settings Parameters (continues on pages 176 to 177)

Parameter	Description
Secured Web Connection **[HTTPSOnly]**	Determines the protocol types used to access the Embedded Web Server. HTTP and HTTPS **[0]** (default). HTTPS only **[1]** (unencrypted HTTP packets are blocked).
HTTP Authentication Mode **[WebAuthMode]**	Determines the authentication mode for the Embedded Web Server. Basic **[0]** = Basic authentication (clear text) is used (default). Digest When Possible **[1]** = Digest authentication (MD5) is used. Basic if HTTPS, Digest if HTTP **[2]** = Digest authentication (MD5) is used for HTTP, and basic authentication is used for HTTPS. Note that when RADIUS login is enabled (WebRADIUSLogin = 1), basic authentication is forced.
Voice Menu Password **[VoiceMenuPassword]**	Password for the voice menu, used for configuration and status. To activate the menu, connect an analog telephone and dial *** (three stars) followed by the password. The default value is 12345. For detailed information on the voice menu, refer to Section 4.2.3 on page 49.
RADIUS General Settings	
Enable RADIUS Access Control **[EnableRADIUS]**	Enables / disables the RADIUS application. Disable **[0]** = RADIUS application is disabled (default). Enable **[1]** = RADIUS application is enabled.
Use RADIUS for Web/Telnet Login **[WebRADIUSLogin]**	Uses RADIUS queries for Web and Telnet interface authentication. Disable **[0]** (default). Enable **[1]**. When enabled, logging to the gateway's Web and Telnet embedded servers is performed via a RADIUS server. The gateway contacts a predefined server and verifies the given username and password pair against a remote database, in a secure manner. **Note 1:** The parameter 'EnableRADIUS' must be set to 1. **Note 2:** RADIUS authentication requires HTTP basic authentication, meaning the username and password are transmitted in clear text over the network. Therefore, users are recommended to set the parameter 'HttpsOnly = 1' to force the use of HTTPS, since the transport is encrypted.
RADIUS Authentication Server IP Address **[RADIUSAuthServerIP]**	IP address of the RADIUS authentication server.
RADIUS Authentication Server Port **[RADIUSAuthPort]**	Port number of the RADIUS authentication server. The default value is 1645.
RADIUS Shared Secret **[SharedSecret]**	'Secret' used to authenticate the gateway to the RADIUS server. Should be a cryptographically strong password.
RADIUS Authentication Settings	
Default Access Level **[DefaultAccessLevel]**	Defines the default access level for the gateway when the RADIUS (authentication) response doesn't include an access level attribute. The valid range is 0 to 255. The default value is 200 (Security Administrator').
Device Behavior Upon RADIUS Timeout **[BehaviorUponRadiusTimeout]**	Defines the gateway's operation if a response isn't received from the RADIUS server after the 5 seconds timeout expires: Deny Access **[0]** = the gateway denies access to the Web and Telnet embedded servers. Verify Access Locally **[1]** = the gateway checks the local username and password (default).

Table 5-36: General Security Settings Parameters (continues on pages 176 to 177)

Parameter	Description
Local RADIUS Password Cache Mode **[RadiusLocalCacheMode]**	Defines the gateway's mode of operation regarding the timer (configured by the parameter RadiusLocalCacheTimeout) that determines the validity of the username and password (verified by the RADIUS server). Absolute Expiry Timer **[0]** = when you access a Web screen, the timeout doesn't reset but rather continues decreasing. Reset Timer Upon Access **[1]** = upon each access to a Web screen, the timeout always resets (reverts to the initial value configured by RadiusLocalCacheTimeout).
Local RADIUS Password Cache Timeout **[RadiusLocalCacheTimeout]**	Defines the time (in seconds) the locally stored username and password (verified by the RADIUS server) are valid. When this time expires, the username and password becomes invalid and a must re-verified with the RADIUS server. The valid range is 1 to 0xFFFFFF. -1 = Never expires. 0 = Each request requires RADIUS authentication. The default value is 300 (5 minutes).
RADIUS VSA Vendor ID **[RadiusVSAVendorID]**	Defines the vendor ID the gateway accepts when parsing a RADIUS response packet. The valid range is 0 to 0xFFFFFFFF. The default value is 5003.
RADIUS VSA Access Level Attribute **[RadiusVSAAccessAttribute]**	Defines the code that indicates the access level attribute in the Vendor Specific Attributes (VSA) section of the received RADIUS packet. The valid range is 0 to 255. The default value is 35.
EtherDiscover Settings	
EtherDiscover Operation Mode	N/A.
SRTP Settings	
Enable Media Security **[EnableMediaSecurity]**	Enables or disables the Secure Real-Time Transport Protocol (SRTP). 0 = SRTP is disabled (default). 1 = SRTP is enabled. **Note 1:** SRTP is available only if DSPVersionTemplateNumber = 0 or 2. **Note 2:** Use of SRTP has the following limitations. For digital 4-span modules, the channels are reduced from 120 to 100. For analog modules with four ports, the 4th port (right-most port on the module) is disabled.
Media Security Behavior **[MediaSecurityBehaviour]**	Determines the gateway's mode of operation when SRTP is used (EnableMediaSecurity = 1). **[0]** (Preferable) = The gateway initiates encrypted calls. If negotiation of the cipher suite fails, an unencrypted call is established. Incoming calls that don't include encryption information are accepted. **[1]** (Mandatory) = The gateway initiates encrypted calls. If negotiation of the cipher suite fails, the call is terminated. Incoming calls that don't include encryption information are rejected (default).
IPSec Settings	
Enable IP Security **[EnableIPSec]**	Enables / disables the Secure Internet Protocol (IPSec) on the gateway. Disable **[0]** = IPSec is disabled (default). Enable **[1]** = IPSec is enabled.

5.6.7.6 Configuring the IPSec Table

Use the IPSec Table screen to configure the IPSec parameters. For detailed information on IPSec and IKE, refer to Section 14.1 on page 343.

5.6.7.7 Configuring the IKE Table

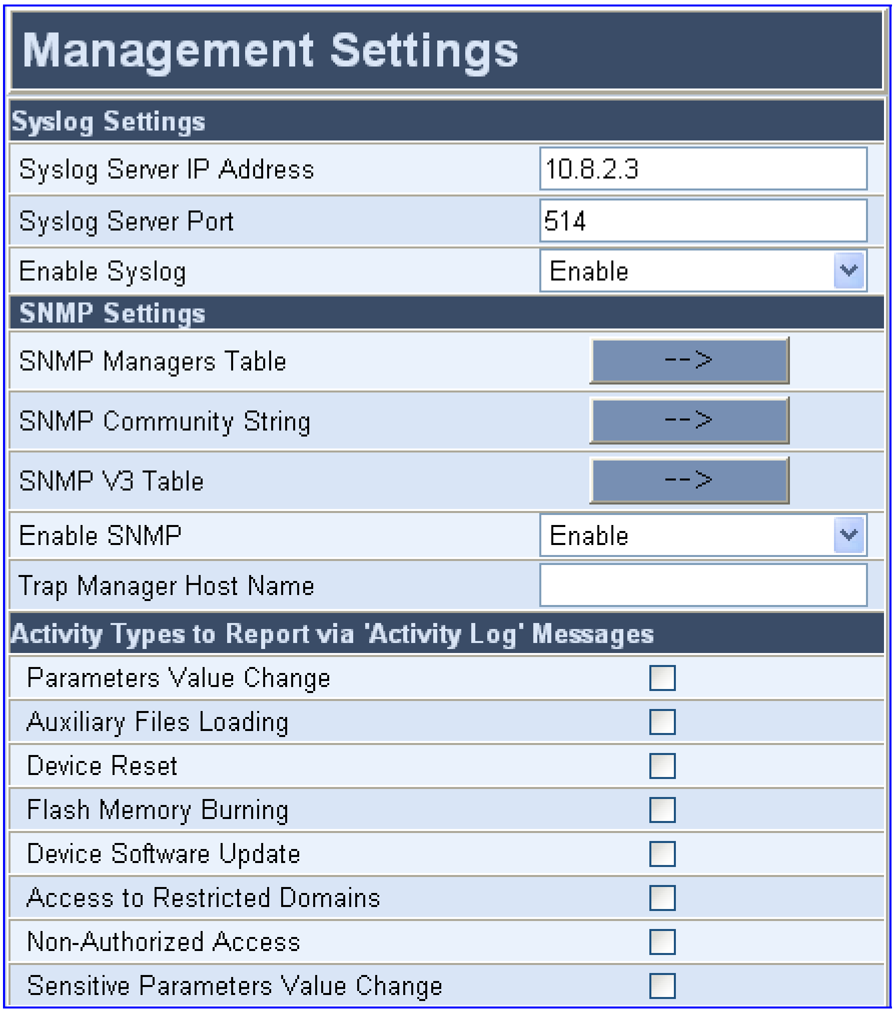
Use the IKE Table screen to configure the IKE parameters. For detailed information on IPSec and IKE, refer to Section 14.1 on page 343.

5.6.8 Configuring the Management Settings

➢ **To configure the Management Settings parameters, take these 4 steps:**

1. Open the 'Management Settings' screen (**Advanced Configuration** menu > **Management Settings**); the 'Management Settings' screen is displayed.

Figure 5-54: Management Settings Screen



Management Settings

Syslog Settings	
Syslog Server IP Address	10.8.2.3
Syslog Server Port	514
Enable Syslog	Enable

SNMP Settings	
SNMP Managers Table	-->
SNMP Community String	-->
SNMP V3 Table	-->
Enable SNMP	Enable
Trap Manager Host Name	

Activity Types to Report via 'Activity Log' Messages	
Parameters Value Change	□
Auxiliary Files Loading	□
Device Reset	□
Flash Memory Burning	□
Device Software Update	□
Access to Restricted Domains	□
Non-Authorized Access	□
Sensitive Parameters Value Change	□

2. Configure the Management Settings according to Table 5-37.
3. Click the **Submit** button to save your changes.
4. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Table 5-37: Management Settings Parameters (continues on pages 179 to 180)

Parameter	Description
Syslog Settings	
Syslog Server IP address **[SyslogServerIP]**	IP address (in dotted format notation) of the computer you are using to run the Syslog Server. The Syslog Server is an application designed to collect the logs and error messages generated by the VoIP gateway. **Note:** Use the SyslogServerPort parameter to define the Syslog server's port. For information on the Syslog, refer to Section 15.2 on page 366.
Syslog Server Port **[SyslogServerPort]**	Defines the UDP port of the Syslog server. The valid range is 0 to 65,535. The default port value is 514. For information on the Syslog, refer to Section 15.2 on page 366.
Enable Syslog **[EnableSyslog]**	Sends the logs and error message generated by the gateway to the Syslog Server. 0 = Disable (logs and errors are not sent to the Syslog Server -- default). 1 = Enable. **Note1:** If you enable Syslog (i.e, EnableSyslog = 1), you must enter an IP address and a port number using SyslogServerIP and SyslogServerPort parameters. **Note 2:** Syslog messages may increase the network traffic. **Note 3:** To configure the Syslog logging levels use the parameter 'GwDebugLevel'. **Note 4:** Logs are also sent to the RS-232 serial port (for information on establishing a serial communications link with the Mediant 1000, refer to Section 12.2 on page 319). For information on the Syslog, refer to Section 15.2 on page 366.
SNMP Settings	
For detailed information on the SNMP parameters that can be configured via the *ini* file, refer to Section 6.5.6 on page 229. For detailed information on developing an SNMP-based program to manage your devices, refer to Section 16 on page 369.	
SNMP Managers Table	Refer to Section 5.6.8.1 on page 180.
SNMP Community Strings	Refer to Section 5.6.8.2 on page 181.
Enable SNMP **[DisableSNMP]**	Enable **[0]** = SNMP is enabled (default). Disable **[1]** = SNMP is disabled and no traps are sent.
Trap Manager Host Name **[SNMPTrapManagerHostName]**	Defines a FQDN of a remote host that is used as an SNMP Manager. The resolved IP address replaces the last entry in the trap manager table (defined by the parameter 'SNMPManagerTableIP_x') and the last trap manager entry of snmpTargetAddrTable in the snmpTargetMIB. For example: 'mngr.corp.mycompany.com'. The valid range is a 99-character string
Activity Types to Report via 'Activity Log' Messages The Activity Log mechanism enables the Mediant 1000 to send log messages (to a Syslog server) that report certain types of web actions according to a pre-defined filter. The following filters are available:	
Parameters Value Change **[ActivityListToLog = PVC]**	Changes made on-the-fly to parameters.
Auxiliary Files Loading **[ActivityListToLog = AFL]**	Loading of auxiliary files (e.g., via Certificate screen).
Device Reset **[ActivityListToLog = DR]**	Device reset via the 'Maintenance Actions' screen.
Flash Memory Burning **[ActivityListToLog = FB]**	Burning of files / parameters to flash (e.g., 'Maintenance Actions' screen).

Table 5-37: Management Settings Parameters (continues on pages 179 to 180)

Parameter	Description
Device Software Update **[ActivityListToLog = SWU]**	*cmp* loading via the Software Upgrade Wizard.
Access to Restricted Domains **[ActivityListToLog = ARD]**	Access to Restricted Domains. The following screens are restricted: (1) *ini* parameters (AdminPage) (2) General Security Settings (3) Configuration File (4) IPSec/IKE tables (5) Software Upgrade Key (6) Internal Firewall (7) Web Access List. (8) Web User Accounts
Non-Authorized Access **[ActivityListToLog = NAA]**	Attempt to access the Embedded Web Server with a false / empty username or password.
Sensitive Parameters Value Change **[ActivityListToLog = SPC]**	Changes made to sensitive parameters: (1) IP Address (2) Subnet Mask (3) Default Gateway IP Address (4) ActivityListToLog

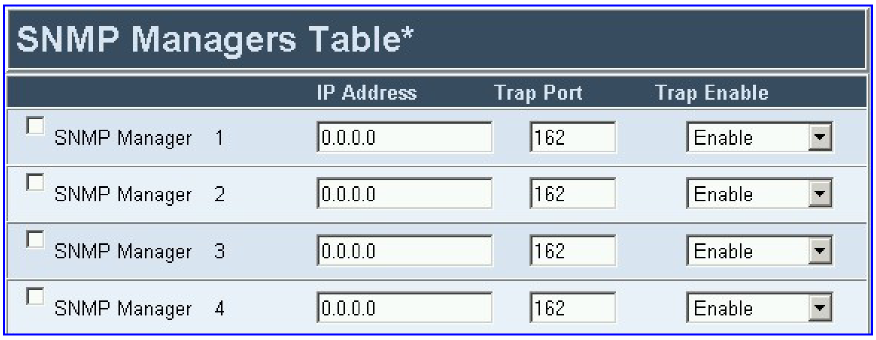
5.6.8.1 Configuring the SNMP Managers Table

The SNMP Managers table allows you to configure the attributes of up to five SNMP managers.

➢ To configure the SNMP Managers Table, take these 6 steps:

1. Access the ‘Management Settings’ screen (**Advanced Configuration** menu > **Management Settings**); the ‘Management Settings’ screen is displayed (Figure 5-54).
2. Open the 'SNMP Managers Table' screen by clicking the arrow sign (-->) to the right of the SNMP Managers Table label; the 'SNMP Managers Table' screen is displayed (Figure 5-55).

Figure 5-55: SNMP Managers Table Screen



SNMP Managers Table*

	IP Address	Trap Port	Trap Enable
□ SNMP Manager 1	0.0.0.0	162	Enable
□ SNMP Manager 2	0.0.0.0	162	Enable
□ SNMP Manager 3	0.0.0.0	162	Enable
□ SNMP Manager 4	0.0.0.0	162	Enable

3. Configure the SNMP Managers parameters according to Table 5-38 on page 181.
4. Click the **Submit** button to save your changes.
5. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Note: If you clear a checkbox and then click **Submit**, all settings in the same row revert to their defaults.

Table 5-38: SNMP Managers Table Parameters

Parameter	Description
Checkbox **[SNMPManagerIsUsed_x]**	Up to five parameters, each determines the **validity** of the parameters (IP address and port number) of the corresponding SNMP Manager used to receive SNMP traps. Checkbox cleared **[0]** = Disabled (default) Checkbox selected **[1]** = Enabled
IP Address **[SNMPManagerTableIP_x]**	Up to five IP addresses of remote hosts that are used as SNMP Managers. The device sends SNMP traps to these IP addresses. Enter the IP address in dotted format notation, for example 108.10.1.255.
Trap Port **[SNMPManagerTrapPort_x]**	Up to five parameters used to define the Port numbers of the remote SNMP Managers. The device sends SNMP traps to these ports. **Note:** The first entry (out of the five) replaces the obsolete parameter SNMPTrapPort. The default SNMP trap port is 162 The valid SNMP trap port range is 100 to 4000.
Trap Enable **[SNMPManagerTrapSending Enable_x]**	Up to five parameters, each determines the activation/deactivation of sending traps to the corresponding SNMP Manager. Disable **[0]** = Sending is disabled Enable **[1]** = Sending is enabled (default)

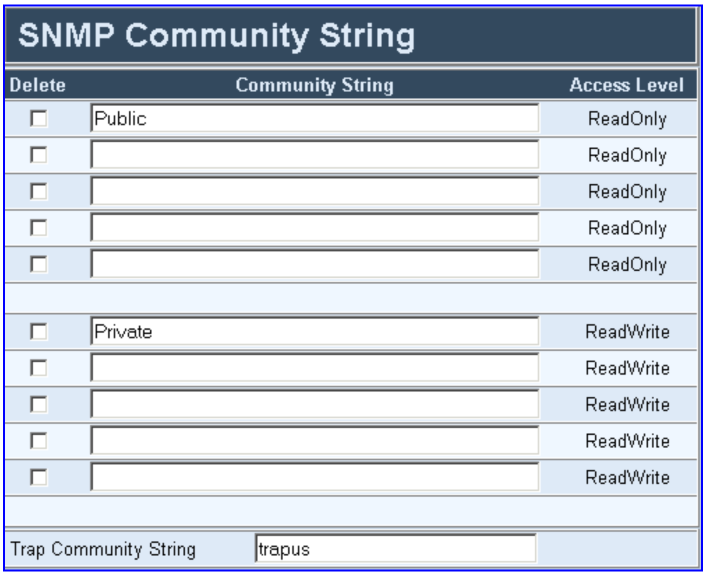
5.6.8.2 Configuring the SNMP Community Strings

Use the SNMP Community Strings table to configure up to five read-only and up to five read / write SNMP community strings, and to configure the community string that is used for sending traps. For detailed information on SNMP community strings, refer to section 16.7.1 on page 377.

➢ To configure the SNMP Community Strings, take these 5 steps:

1. Access the ‘Management Settings’ screen (**Advanced Configuration** menu > **Management Settings**); the ‘Management Settings’ screen is displayed (Figure 5-54).
2. Open the 'SNMP Community Strings' screen by clicking the arrow sign (-->) to the right of the SNMP Community Strings label; the 'SNMP Community Strings' screen is displayed (Figure 5-56).

Figure 5-56: SNMP Community Strings Screen



SNMP Community String

Delete	Community String	Access Level
□	Public	ReadOnly
□		ReadOnly
□		ReadOnly
□		ReadOnly
□		ReadOnly
□	Private	ReadWrite
□		ReadWrite
□		ReadWrite
□		ReadWrite
□		ReadWrite

Trap Community String: trapus

3. Configure the SNMP Community Strings parameters according to Table 5-39.
4. Click the **Submit** button to save your changes.
5. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

Note: To delete a community string, select the **Delete** checkbox to the left of the community string you want to delete, and then click the button **Submit**.

Table 5-39: SNMP Community Strings Parameters

Parameter	Description
Read Only Community String **[SNMPReadOnlyCommunityString_x]**	Up to five read-only community strings (up to 19 characters each). The default string is 'public'.
Read / Write Community String **[SNMPReadWriteCommunityString_x]**	Up to five read / write community strings (up to 19 characters each). The default string is 'private'.
Trap Community String **[SNMPTrapCommunityString]**	Community string used in traps (up to 19 characters). The default string is 'trapuser'.

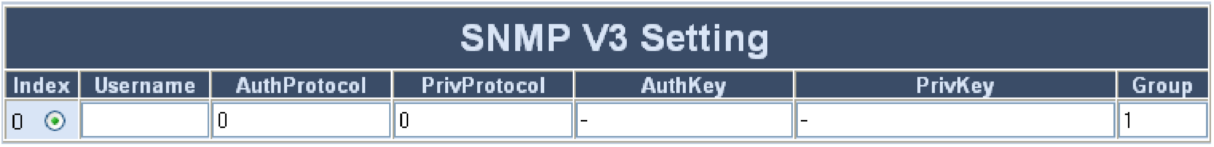
5.6.8.3 Configuring SNMP V3 Users

Use the SNMP V3 Table to configure authentication and privacy for up to 10 SNMP v3 users.

➢ To configure the SNMP v3 users, take the following 6 steps:

1. Access the 'Management Settings' screen (**Advanced Configuration** menu > **Management Settings**); the 'Management Settings' screen is displayed.
2. Open the 'SNMP V3 Setting' screen by clicking the arrow sign (-->) to the right of the SNMP V3 Table label; the 'SNMP V3 Setting' screen is displayed (Figure 5-57).

Figure 5-57: SNMP V3 Setting Screen



SNMP V3 Setting

Index	Username	AuthProtocol	PrivProtocol	AuthKey	PrivKey	Group
0 ◉		0	0	-	-	1

3. To add an SNMP v3 user, in the 'New Row Index' field, type the desired row index, and then click **Add an Empty Row**. A new row appears.
4. Configure the SNMP V3 Setting parameters according to Table 5-40 on page 183.
5. Click the **Apply Row Settings** button to save your changes.
6. To save the changes so they are available after a hardware reset or power fail, refer to Section 5.9.2 on page 200.

Notes:

- To delete an SNMP V3 user, select the Index radio button corresponding to the SNMP V3 user row entry that you want to delete, and then click the **Delete Row** button.
- To copy an existing SNMP V3 user configuration to a new row, select the radio button on the left of the desired SNMP V3 user row. and then click **Copy Selected Row as A New Row**. A new row appears that includes the same configuration as the selected row.
- To sort all row indexes incrementally, click **Compact Table**.

Table 5-40: SNMP V3 Users Parameters (continues on pages 183 to 183)

Parameter	Description
SNMPUsers_Index [Index]	This is the table index. Its valid range is 0 to 9.
[SNMPUsers_Username] [Username]	Name of the SNMP v3 user. This name must be unique.
[SNMPUsers_AuthProtocol] [AuthProtocol]	Authentication protocol to be used for the SNMP v3 user. 0 = none (default) 1 = MD5 2 = SHA-1
[SNMPUsers_PrivProtocol] [PrivProtocol]	Privacy protocol to be used for the SNMP v3 user. 0 = none (default) 1 = DES 2 = 3DES 3 = AES128 4 = AES192 5 = AES256
[SNMPUsers_AuthKey] [AuthKey]	Authentication key. Keys can be entered in the form of a text password or long hex string. Keys are always persisted as long hex strings and keys are localized.
[SNMPUsers_PrivKey] [PrivKey]	Privacy key. Keys can be entered in the form of a text password or long hex string. Keys are always persisted as long hex strings and keys are localized.
[SNMPUsers_Group] [Group]	The group with which the SNMP v3 user is associated. 0 = read-only group (default) 1 = read-write group 2 = trap group **Note:** all groups can be used to send traps.

5.7 Status & Diagnostics

Use this menu to view and monitor the gateway's channels, Syslog messages, hardware / software product information, and to assess the gateway's statistics and IP connectivity information.

5.7.1 Gateway Statistics

Use the screens under Gateway Statistics to monitor real-time activity such as IP Connectivity information, call details and call statistics, including the number of call attempts, failed calls, fax calls, etc.

Note: The Gateway Statistics screens don't refresh automatically. To view updated information, re-access the screen you require.

5.7.1.1 IP Connectivity

The IP Connectivity screen provides you with online read-only network diagnostic connectivity information on all destination IP addresses configured in the Tel to IP Routing table (refer to Section 5.5.4.2 on page 107).

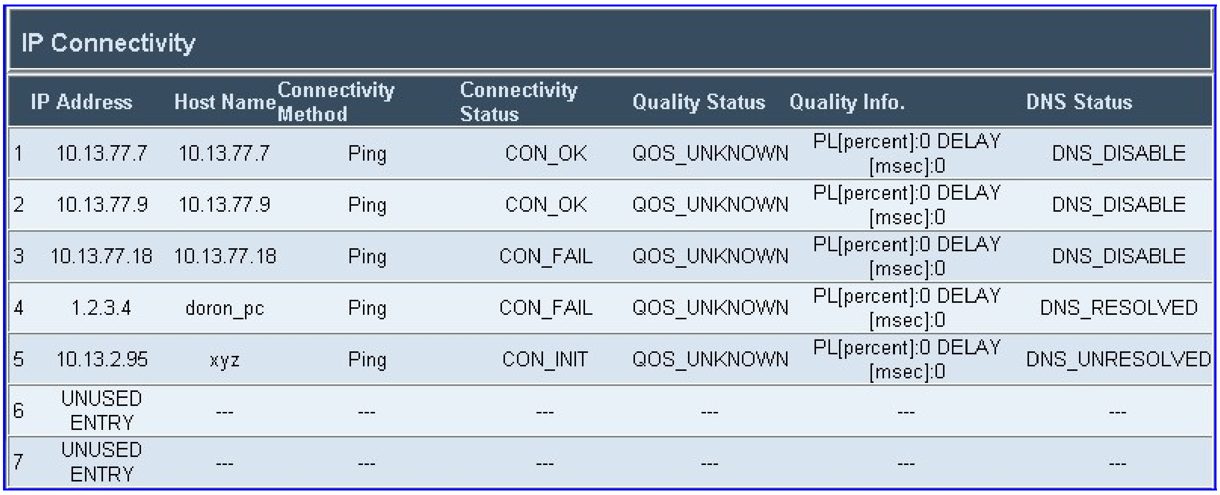
Notes:

- This information is available only if the parameter AltRoutingTel2IPEnable (described in Table 5-14) is set to 1 (Enable) or 2 (Status Only).
- The information in columns 'Quality Status' and 'Quality Info.' (per IP address) is reset if two minutes elapse without a call to that destination.

➢ To view the IP connectivity information, take these 2 steps:

1. Set the parameter 'Enable Alt Routing Tel to IP' (or *ini* file parameter AltRoutingTel2IPEnable) to 'Enable' (1) or 'Status Only' (2). To configure this parameter, refer to Section 5.5.4.1 on page 105.
2. Open the 'IP Connectivity' screen (**Status & Diagnostics** menu > **Gateway Statistics** submenu > **IP Connectivity**); the 'IP Connectivity' screen is displayed (Figure 5-58).

Figure 5-58: IP Connectivity Screen



IP Connectivity

	IP Address	Host Name	Connectivity Method	Connectivity Status	Quality Status	Quality Info.	DNS Status
1	10.13.77.7	10.13.77.7	Ping	CON_OK	QOS_UNKNOWN	PL[percent]:0 DELAY [msec]:0	DNS_DISABLE
2	10.13.77.9	10.13.77.9	Ping	CON_OK	QOS_UNKNOWN	PL[percent]:0 DELAY [msec]:0	DNS_DISABLE
3	10.13.77.18	10.13.77.18	Ping	CON_FAIL	QOS_UNKNOWN	PL[percent]:0 DELAY [msec]:0	DNS_DISABLE
4	1.2.3.4	doron_pc	Ping	CON_FAIL	QOS_UNKNOWN	PL[percent]:0 DELAY [msec]:0	DNS_RESOLVED
5	10.13.2.95	xyz	Ping	CON_INIT	QOS_UNKNOWN	PL[percent]:0 DELAY [msec]:0	DNS_UNRESOLVED
6	UNUSED ENTRY	---	---	---	---	---	---
7	UNUSED ENTRY	---	---	---	---	---	---

Table 5-41: IP Connectivity Parameters

Column Name	Description
IP Address	IP address defined in the destination IP address field in the Tel to IP Routing table. - or - IP address that is resolved from the host name defined in the destination IP address field in the Tel to IP Routing table.
Host Name	Host name (or IP address) defined in the destination IP address field in the Tel to IP Routing table.
Connectivity Method	The method according to which the destination IP address is queried periodically (currently only by ping).
Connectivity Status	Displays the status of the IP address' connectivity according to the method in the 'Connectivity Method' field. Can be one of the following: ▪ OK = Remote side responds to periodic connectivity queries. ▪ Lost = Remote side didn't respond for a short period. ▪ Fail = Remote side doesn't respond. ▪ Init = Connectivity queries not started (e.g., IP address not resolved). ▪ Disable = The connectivity option is disabled ('AltRoutingTel2IPMode' equals 0 or 2).
Quality Status	Determines the QoS (according to packet loss and delay) of the IP address. Can be one of the following: ▪ Unknown = Recent quality information isn't available. ▪ OK ▪ Poor **Note 1:** This field is applicable only if the parameter 'AltRoutingTel2IPMode' is set to 2 or 3. **Note 2:** This field is reset if no QoS information is received for 2 minutes.
Quality Info.	Displays QoS information: delay and packet loss, calculated according to previous calls. **Note 1:** This field is applicable only if the parameter 'AltRoutingTel2IPMode' is set to 2 or 3. **Note 2:** This field is reset if no QoS information is received for 2 minutes.
DNS Status	Can be one of the following: ▪ DNS Disable ▪ DNS Resolved ▪ DNS Unresolved

5.7.1.2 Call Counters

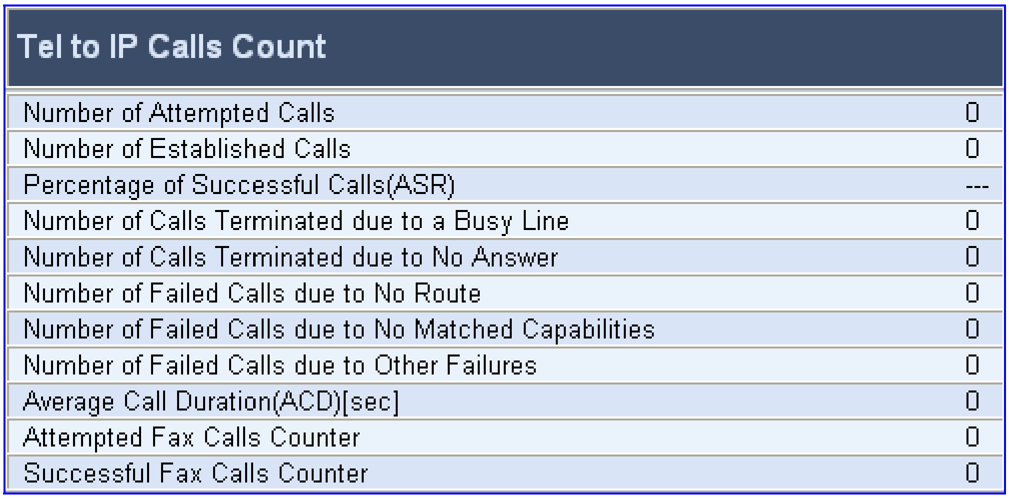
The Call Counters screens provide you with statistic information on incoming (IP→Tel) and outgoing (Tel→IP) calls. The statistic information is updated according to the release reason that is received after a call is terminated (during the same time as the end-of-call CDR message is sent). The release reason can be viewed in the Termination Reason field in the CDR message. For detailed information on each counter, refer to Table 5-42 on page 186.

You can reset this information by clicking the **Reset Counters** button.

➢ To view the IP→Tel and Tel→IP Call Counters information, take this step:

- Open the Call Counters screen you want to view (**Status & Diagnostics** menu > **Gateway Statistics** submenu > **IP to Tel Calls Count** or **Tel to IP Calls Count** option); the relevant Call Counters screen is displayed. Figure 5-59 shows the 'Tel→IP Call Counters' screen.

Figure 5-59: Tel to IP Calls Count Screen



Tel to IP Calls Count

Number of Attempted Calls	0
Number of Established Calls	0
Percentage of Successful Calls(ASR)	---
Number of Calls Terminated due to a Busy Line	0
Number of Calls Terminated due to No Answer	0
Number of Failed Calls due to No Route	0
Number of Failed Calls due to No Matched Capabilities	0
Number of Failed Calls due to Other Failures	0
Average Call Duration(ACD)[sec]	0
Attempted Fax Calls Counter	0
Successful Fax Calls Counter	0

Table 5-42: Call Counters Description (continues on pages 186 to 187)

Counter	Description
Number of Attempted Calls	This counter indicates the number of attempted calls. It is composed of established and failed calls. The number of established calls is represented by the 'Number of Established Calls' counter. The number of failed calls is represented by the five failed-call counters. Only one of the established / failed call counters is incremented every time.

Table 5-42: Call Counters Description (continues on pages 186 to 187)

Counter	Description
Number of Established Calls	This counter indicates the number of established calls. It is incremented as a result of one of the following release reasons, if the duration of the call is bigger then zero: GWAPP_REASON_NOT_RELEVANT (0) GWAPP_NORMAL_CALL_CLEAR (16) GWAPP_NORMAL_UNSPECIFIED (31) And the internal reasons: RELEASE_BECAUSE_UNKNOWN_REASON RELEASE_BECAUSE_REMOTE_CANCEL_CALL RELEASE_BECAUSE_MANUAL_DISC RELEASE_BECAUSE_SILENCE_DISC RELEASE_BECAUSE_DISCONNECT_CODE **Note:** When the duration of the call is zero, the release reason GWAPP_NORMAL_CALL_CLEAR increments the 'Number of Failed Calls due to No Answer' counter. The rest of the release reasons increment the 'Number of Failed Calls due to Other Failures' counter.
Percentage of Successful Calls (ASR)	The percentage of established calls from attempted calls.
Number of Calls Terminated due to a Busy Line	This counter indicates the number of calls that failed as a result of a busy line. It is incremented as a result of the following release reason: GWAPP_USER_BUSY (17)
Number of Calls Terminated due to No Answer	This counter indicates the number of calls that weren't answered. It is incremented as a result of one of the following release reasons: GWAPP_NO_USER_RESPONDING (18) GWAPP_NO_ANSWER_FROM_USER_ALERTED (19) And (when the call duration is zero) as a result of the following: GWAPP_NORMAL_CALL_CLEAR (16)
Number of Failed Calls due to No Route	This counter indicates the number of calls whose destinations weren't found. It is incremented as a result of one of the following release reasons: GWAPP_UNASSIGNED_NUMBER (1) GWAPP_NO_ROUTE_TO_DESTINATION (3)
Number of Failed Calls due to No Matched Capabilities	This counter indicates the number of calls that failed due to mismatched gateway capabilities. It is incremented as a result of an internal identification of capability mismatch. This mismatch is reflected to CDR via the value of the parameter 'DefaultReleaseReason' (default is GWAPP_NO_ROUTE_TO_DESTINATION (3)), or by the GWAPP_SERVICE_NOT_IMPLEMENTED_UNSPECIFIED(79) reason.
Number of Failed Calls due to Other Failures	This counter is incremented as a result of calls that fail due to reasons not covered by the other counters.
Average Call Duration (ACD) [sec]	The average call duration of established calls.
Attempted Fax Calls Counter	This counter indicates the number of attempted fax calls.
Successful Fax Calls Counter	This counter indicates the number of successful fax calls.

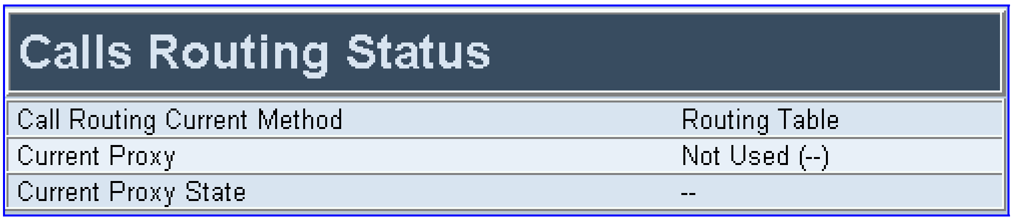
5.7.1.3 Call Routing Status

The Call Routing Status screen provides you with information on the current routing method used by the gateway. This information includes the IP address and FQDN (if used) of the Proxy server with which the gateway currently operates.

➢ To view the Call Routing Status screen, take this step:

- Open the 'Call Routing Status' screen (**Status & Diagnostics** menu > **Gateway Statistics** submenu > **Calls Routing Status** option); the Call Routing Status screen is displayed.

Figure 5-60: Call Routing Status Screen



Calls Routing Status

Call Routing Current Method	Routing Table
Current Proxy	Not Used (--)
Current Proxy State	--

Table 5-43: Call Routing Status Parameters

Parameter	Description
Current Call-Routing Method	Proxy = Proxy server is used to route calls.
	Routing Table preferred to Proxy = The Tel to IP Routing table takes precedence over a Proxy for routing calls (PreferRouteTable = 1).
	Routing Table = The Tel to IP Routing table is used to route calls.
Current Proxy	Not Used = Proxy server isn't defined.
	IP address and FQDN (if exists) of the Proxy server the gateway currently operates with.
Current Proxy State	N/A = Proxy server isn't defined.
	OK = Communication with the Proxy server is in order.
	Fail = No response from any of the defined Proxies.

5.7.2 Activating the Internal Syslog Viewer

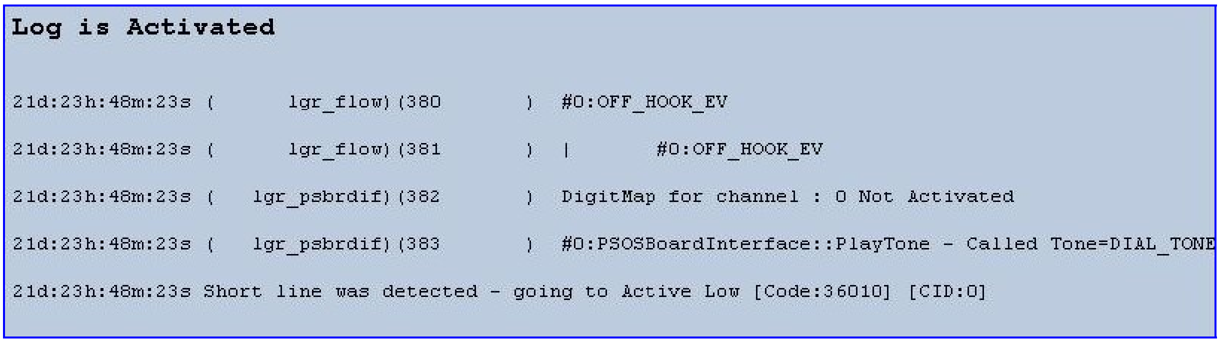
The Message Log screen displays Syslog debug messages sent by the gateway. You can simply select the messages, and then copy and paste them into a text editor such as Notepad. This text file (*txt*) can then be sent to our Technical Support for diagnosis and troubleshooting.

> **Note:** It's not recommended to keep a 'Message Log' session open (even if the window is minimized) for a prolonged period (refer to the Note below). This may cause the Mediant 1000 to overload. For prolong debugging use an external Syslog server (refer to Section 15.3 on page 366).

➢ To activate the Message Log, take these 3 steps:

1. In the 'General Parameters' screen (**Protocol Management** menu > **Advanced Parameters** submenu > **General Parameters** option), set the parameter 'Debug Level' (or *ini* file parameter GwDebugLevel) to 5 (refer to Section 5.5.2.1 on page 83). This parameter determines the Syslog logging level in the range 0 to 5, where 5 is the highest level.
2. Open the 'Message Log' screen (**Status & Diagnostics** menu > **Message Log**); the 'Message Log' screen is displayed and the Log is activated.

Figure 5-61: Message Log Screen



```
Log is Activated

21d:23h:48m:23s (      lgr_flow)(380       )  #0:OFF_HOOK_EV

21d:23h:48m:23s (      lgr_flow)(381       )  |       #0:OFF_HOOK_EV

21d:23h:48m:23s (   lgr_psbrdif)(382       )  DigitMap for channel : 0 Not Activated

21d:23h:48m:23s (   lgr_psbrdif)(383       )  #0:PSOSBoardInterface::PlayTone - Called Tone=DIAL_TONE

21d:23h:48m:23s Short line was detected - going to Active Low [Code:36010] [CID:0]
```

3. To clear the screen of messages, click the submenu **Message Log** again; the screen is cleared and new messages begin appearing.

➢ To de-activate the Message Log, take this step:

- Close the screen by accessing any another screen.

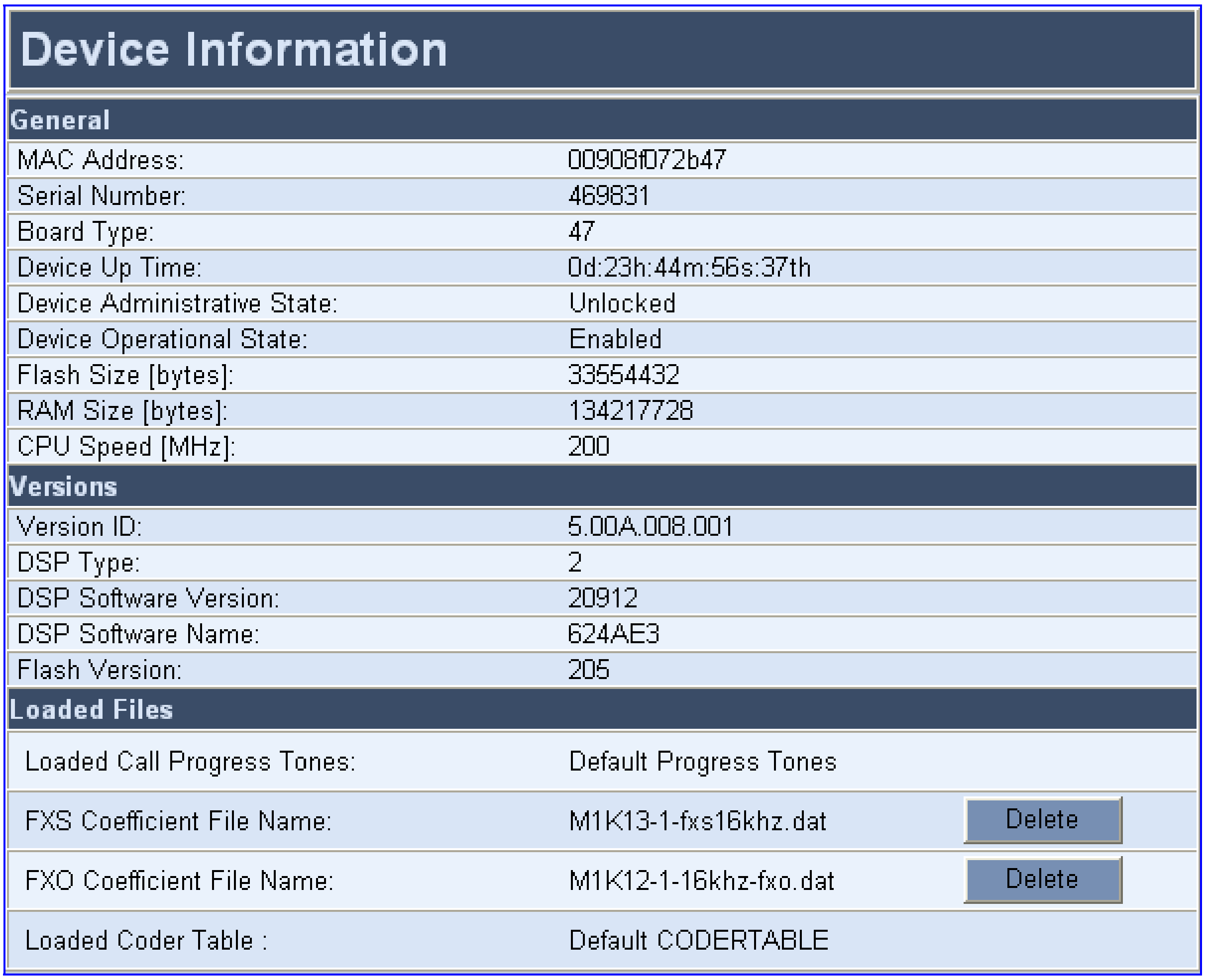
5.7.3 Device Information

The Device Information screen displays specific hardware and software product information. This information can help you to expedite any troubleshooting process. Capture the screen and email it to ‘our’ Technical Support personnel to ensure quick diagnosis and effective corrective action. From this screen you can also view and remove any loaded files used by the Mediant 1000 (stored in the RAM).

➢ **To access the System Information screen, take this step:**

- Open the ‘Device Information’ screen (**Status & Diagnostics** menu > **Device Information**); the ‘Device Information’ screen is displayed.

Figure 5-62: Device Information Screen



Device Information

General		
MAC Address:	00908f072b47	
Serial Number:	469831	
Board Type:	47	
Device Up Time:	0d:23h:44m:56s:37th	
Device Administrative State:	Unlocked	
Device Operational State:	Enabled	
Flash Size [bytes]:	33554432	
RAM Size [bytes]:	134217728	
CPU Speed [MHz]:	200	
Versions		
Version ID:	5.00A.008.001	
DSP Type:	2	
DSP Software Version:	20912	
DSP Software Name:	624AE3	
Flash Version:	205	
Loaded Files		
Loaded Call Progress Tones:	Default Progress Tones	
FXS Coefficient File Name:	M1K13-1-fxs16khz.dat	Delete
FXO Coefficient File Name:	M1K12-1-16khz-fxo.dat	Delete
Loaded Coder Table :	Default CODERTABLE	

➢ **To delete any of the loaded files, take these 3 steps:**

1. Click the **Delete** button to the right of the files you want to delete. Deleting a file takes effect only after the Mediant 1000 is reset.
2. Click the **Maintenance** button on the main menu bar; the 'Maintenance Actions' screen is displayed.
3. From the 'Burn to FLASH' drop-down list, select 'Yes', and then click the **Reset** button; the gateway is shut down and re-activated. A message about the waiting period is displayed.

5.8 Software Update

The Software Update menu enables users to upgrade the Mediant 1000 software by loading a new *cmp* file along with the *ini* and a suite of auxiliary files, or to update the existing auxiliary files.

The ‘Software Update’ menu comprises the following submenus:

- Software Upgrade Wizard (refer to Section 5.8.1 below)
- Load Auxiliary Files (refer to Section 5.8.2 on page 196)
- Software Upgrade Key (refer to section 5.8.3 on page 198)

Note: When upgrading the Mediant 1000 software, you *must* load the new *cmp* file with all other related configuration files.

5.8.1 Software Upgrade Wizard

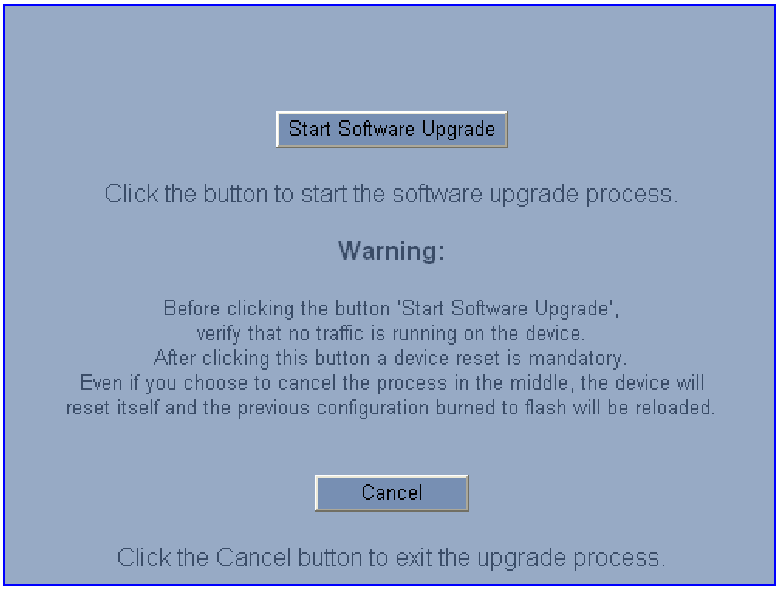
The Software Upgrade Wizard guides users through the process of software upgrade: selecting files and loading them to the gateway. The wizard also enables users to upgrade software while maintaining the existing configuration. Using the wizard obligates users to load and burn a *cmp* file. Users can choose to also use the wizard to load the *ini* and auxiliary files (e.g., Call Progress Tones*)*, but this option cannot be pursued without loading the *cmp* file. For the *ini* and each auxiliary file type, users can choose to reload an existing file, load a new file or not load a file at all.

Warning: The Software Upgrade Wizard requires the Mediant 1000 to be reset at the end of the process, which may disrupt its traffic. To avoid this, disable all traffic on the device before initiating the wizard by performing a graceful lock (refer to Section 5.9.1 on page 198).

➢ To use the Software Upgrade Wizard, take these 11 steps:

1. Stop all traffic on the Mediant 1000 (refer to the note above).
2. Open the ‘Software Upgrade Wizard’ (**Software Update** menu > **Software Upgrade Wizard**); the ‘Start Software Upgrade’ screen appears.

Figure 5-63: Start Software Upgrade Screen



Click the button to start the software upgrade process.

Warning:

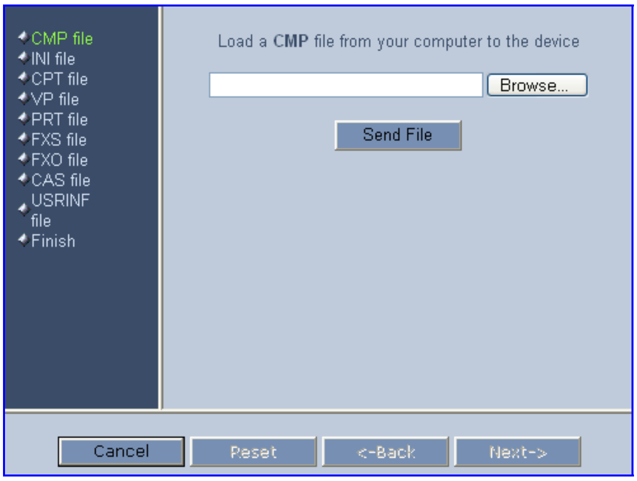
Before clicking the button 'Start Software Upgrade', verify that no traffic is running on the device. After clicking this button a device reset is mandatory. Even if you choose to cancel the process in the middle, the device will reset itself and the previous configuration burned to flash will be reloaded.

Click the Cancel button to exit the upgrade process.

Note: At this point, the process can be canceled with no consequence to the Mediant 1000 (click the **Cancel** button). If you continue the process (by clicking the **Start Software Upgrade** button, the process must be followed through and completed with a Mediant 1000 reset at the end. If you click the **Cancel** button in any of the subsequent screens, the Mediant 1000 is automatically reset with the configuration that was previously burned in flash memory.

4. Click the **Start Software Upgrade** button; the 'Load a cmp file' screen appears (Figure 5-64).

Figure 5-64: Load a *cmp* File Screen

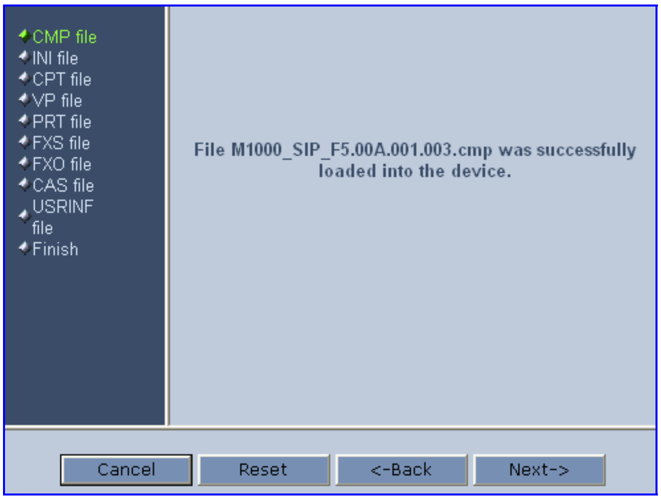


Load a CMP file from your computer to the device

Note: When in the wizard process, the rest of the Web application is unavailable and the background Web screen is disabled. After the process is completed, access to the full Web application is restored.

5. Click the **Browse** button, navigate to the *cmp* file and click the button **Send File**; the *cmp* file is loaded to the Mediant 1000 and you're notified as to a successful loading (refer to Figure 5-65).

Figure 5-65: *cmp* File Successfully Loaded



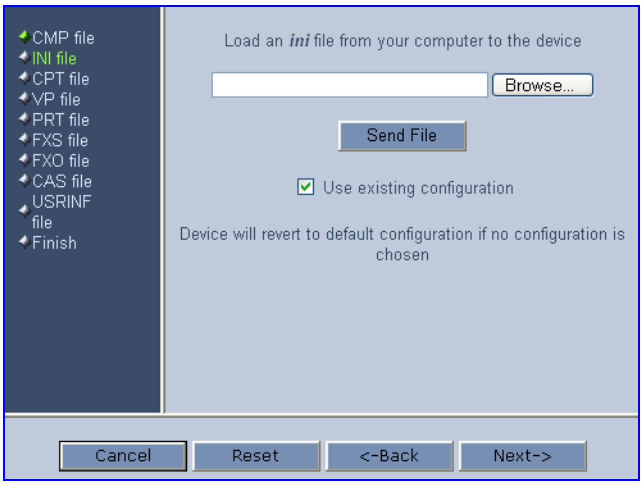
File M1000_SIP_F5.00A.001.003.cmp was successfully loaded into the device.

6. Note that the four action buttons (**Cancel**, **Reset**, **Back**, and **Next**) are now activated (following *cmp* file loading). You can now choose to either:

- Click **Reset**; the Mediant 1000 resets, utilizing the new *cmp* you loaded and utilizing the current configuration files.
- Click **Cancel**; the Mediant 1000 resets utilizing the *cmp, ini* and all other configuration files that were previously stored in flash memory. Note that these are NOT the files you loaded in the previous Wizard steps.
- Click **Back**; the 'Load a *cmp* File' screen is reverted to; refer to Figure 5-64.
- Click **Next**; the 'Load an *ini* File' screen opens; refer to Figure 5-66. Loading a new *ini* file or any other auxiliary file listed in the Wizard is optional.

Note that as you progress, the file type list on the left indicates which file type loading is in process by illuminating green (until 'FINISH').

Figure 5-66: Load an *ini* File Screen



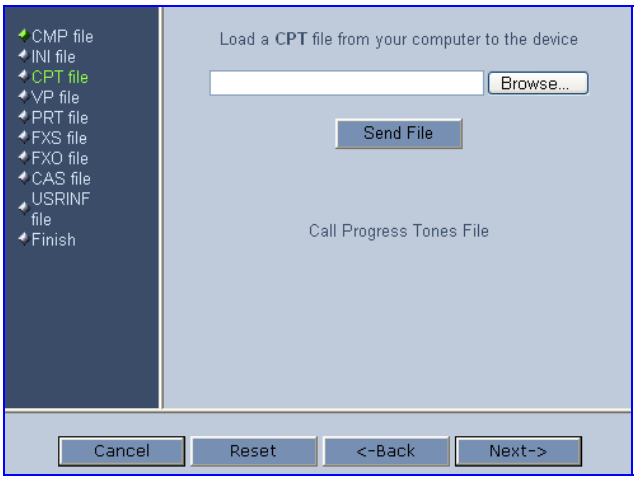
Load an *ini* file from your computer to the device

☑ Use existing configuration

Device will revert to default configuration if no configuration is chosen

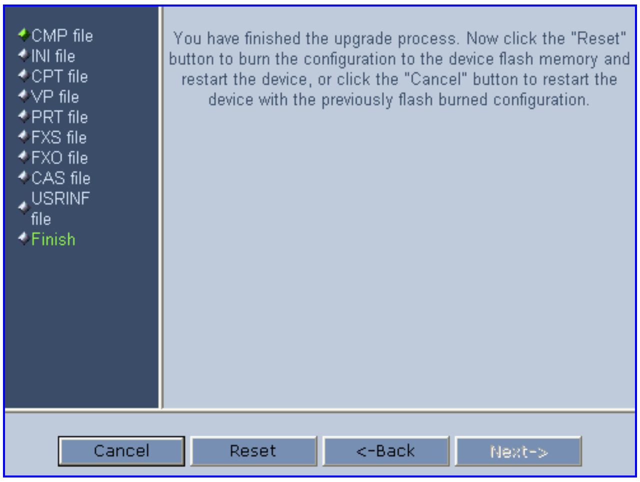
7. In the ‘Load an *ini* File’ screen, you can now choose to either:
   - Click **Browse** and navigate to the *ini* file; the check box ‘Use existing configuration’, by default checked, becomes unchecked. Click **Send File**; the *ini* file is loaded to the Mediant 1000 and you’re notified as to a successful loading.
   - Ignore the **Browse** button (its field remains undefined and the check box ‘Use existing configuration’ remains checked by default).
   - Ignore the **Browse** button and uncheck the ‘Use existing configuration’ check box; no *ini* file is loaded, the Mediant 1000 uses its factory-preconfigured values.
8. You can now choose to either:
   - Click **Cancel**; the Mediant 1000 resets utilizing the *cmp, ini* and all other configuration files that were previously stored in flash memory. Note that these are NOT the files you loaded in the previous Wizard steps.
   - Click **Reset**; the Mediant 1000 resets, utilizing the new *cmp* and *ini* file you loaded up to now as well as utilizing the other configuration files.
   - Click **Back**; the ‘Load a *cmp* file’ screen is reverted to; refer to Figure 5-64.
   - Click **Next**; the ‘Load a CPT File’ screen opens (refer to Figure 5-67); Loading a new CPT file or any other auxiliary file listed in the Wizard is optional.

Figure 5-67: Load a CPT File Screen



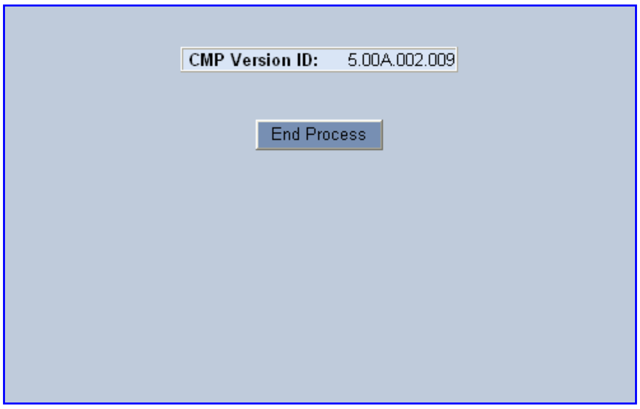
9. Follow the same procedure you followed when loading the *ini* file (refer to Step 7). The same procedure applies to the ‘Load a VP file’ (not applicable to the Mediant 1000 gateway) screen and ‘Load a coefficient file’ screen.
10. In the ‘FINISH’ screen (refer to Figure 5-68), the **Next** button is disabled. Complete the upgrade process by clicking **Reset** or **Cancel**.
    - Click **Reset**, the Mediant 1000 ‘burns’ the newly loaded files to flash memory. The ‘Burning files to flash memory’ screen appears. Wait for the ‘burn’ to finish. When it finishes, the ‘End Process’ screen appears displaying the burned configuration files (refer to Figure 5-69).
    - Click **Cancel**, the Mediant 1000 resets, utilizing the files previously stored in flash memory. (Note that these are NOT the files you loaded in the previous Wizard steps).

Figure 5-68: FINISH Screen



You have finished the upgrade process. Now click the "Reset" button to burn the configuration to the device flash memory and restart the device, or click the "Cancel" button to restart the device with the previously flash burned configuration.

Figure 5-69: 'End Process' Screen



CMP Version ID: 5.00A.002.009

11. Click the **End Process** button; the 'Quick Setup' screen appears and the full Web application is reactivated.

5.8.2 Auxiliary Files

The ‘Auxiliary Files’ screen enables you to load various files to the gateway, as described in Table 5-44. For detailed information on these files, refer to Section 17 on page 391. For information on deleting these files from the Mediant 1000, refer to Section 5.7.3 on page 190. The table presents a brief description of each auxiliary file.

Table 5-44: Auxiliary Files Descriptions

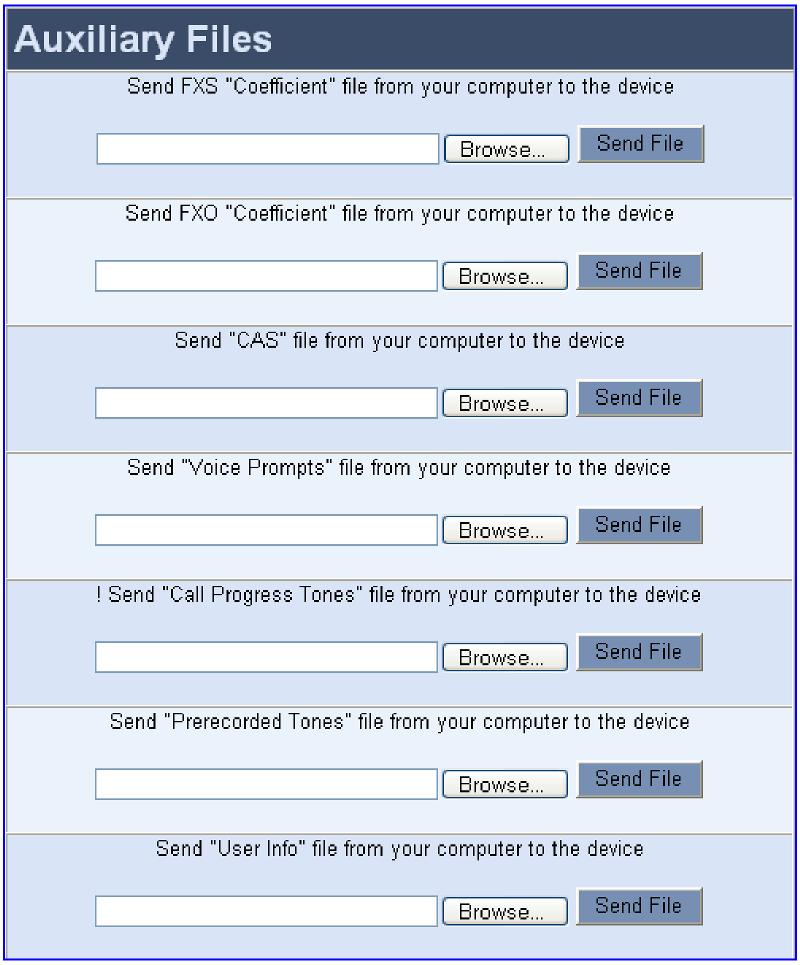
File Type	Description
Coefficient	This file (different file for FXS and FXO modules) contains the telephony interface configuration information for the VoIP gateway. This information includes telephony interface characteristics, such as DC and AC impedance, feeding current and ringing voltage. This file is specific to the type of telephony interface that the VoIP gateway supports. In most cases you have to load this type of file.
CAS	Up to 8 different CAS files containing specific CAS protocol definitions for digital modules. These files are provided to support various types of CAS signaling.
Call Progress Tones	This is a region-specific, telephone exchange-dependent file that contains the Call Progress Tones levels and frequencies that the VoIP gateway uses. The default CPT file is: U.S.A.
Prerecorded Tones	The *dat* PRT file enhances the gateway’s capabilities of playing a wide range of telephone exchange tones that cannot be defined in the Call Progress Tones file.
User Info	The User Information file maps PBX extensions to IP numbers. This file can be used to represent PBX extensions as IP phones in the global ‘IP world’.

5.8.2.1 Loading the Auxiliary Files via the Embedded Web Server

➢ **To load an auxiliary file to the gateway using the embedded Web server, take these 8 steps:**

1. Open the ‘Auxiliary Files’ screen (**Software Upgrade** menu > **Load Auxiliary Files**); the ‘Auxiliary Files’ screen is displayed.

Figure 5-70: Auxiliary Files Screen



Auxiliary Files

Send FXS "Coefficient" file from your computer to the device

Send FXO "Coefficient" file from your computer to the device

Send "CAS" file from your computer to the device

Send "Voice Prompts" file from your computer to the device

! Send "Call Progress Tones" file from your computer to the device

Send "Prerecorded Tones" file from your computer to the device

Send "User Info" file from your computer to the device

2. Click the **Browse** button corresponding to the type of file that you want to load.
3. Navigate to the folder that contains the file you want to load.
4. Select the file, and then click **Open**; the name and path of the file appear in the field next to the **Browse** button.
5. Click the **Send File** button corresponding to the field that contains the name of the file you want to load.
6. Repeat steps 2 through 5 for each file you want to load.
7. To save the loaded auxiliary files so they are available after a power fail, refer to Section 5.9.2 on page 200.
8. To reset the Mediant 1000, refer to Section 5.9.3 on page 200.

Notes:

- Saving an auxiliary file to flash memory may disrupt traffic on the Mediant 1000. To avoid this, disable all traffic on the device by performing a graceful lock (refer to Section 5.9.1 on page 198).
- File names preceded by an exclamation mark (!) are not changeable on-the-fly and require that the device be reset (e.g., Call Progress Tones file).

5.8.2.2 Loading the Auxiliary Files via the *ini* File

Before you load the configuration files (Call Progress Tones, Prerecorded Tones, User Information, FXS/FXO Coefficient, and CAS) to the gateway, in the *ini* file you need to define certain *ini* file parameters associated with these files. These *ini* file parameters specify the files that you want loaded and whether they must be stored in the non-volatile memory. For a description of the *ini* file parameters associated with the configuration files, refer to Section 6.5.14 on page 258.

➢ To load the auxiliary files via the *ini* file, take these 3 steps:

1. In the *ini* file, define the auxiliary files to be loaded to the Mediant 1000. You can also define in the *ini* file whether the loaded files should be stored in the non-volatile memory so that the TFTP process is not required every time the Mediant 1000 boots up. (See note below.)
2. Locate the auxiliary files you want to load and the *ini* file in the same directory.
3. Invoke a BootP/TFTP session; the *ini* and auxiliary files are loaded onto the Mediant 1000.

5.8.3 Updating the Software Upgrade Key

Mediant 1000 gateways are supplied with a Software Upgrade Key already pre-configured for the gateway.

You can later upgrade their Mediant 1000 features, capabilities and quantity of available resources by specifying what upgrades are required, and purchasing a new key to match their specification.

The Software Upgrade Key is sent as a string in a text file, to be loaded into the Mediant 1000. Stored in the device’s non-volatile flash memory, the string defines the features and capabilities allowed by the specific key purchased by the user. The device allows you to utilize *only these* features and capabilities. A new key overwrites a previously installed key.

For detailed information on the Software Upgrade Key, refer to Section 12.7 on page 334.

5.9 Maintenance

The Maintenance menu is used for the following operations:

- Locking and unlocking the gateway (refer to Section 5.9.1 on page 198)
- Saving the gateway's configuration (refer to Section 5.9.2 on page 200)
- Resetting the gateway (refer to Section 5.9.3 on page 200)

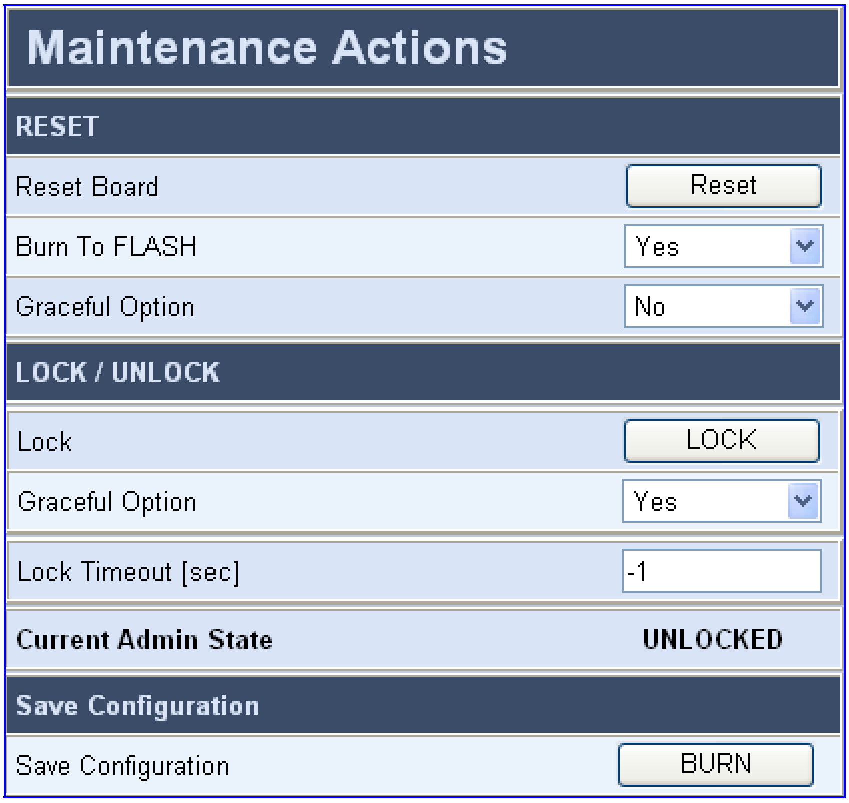
5.9.1 Locking and Unlocking the Gateway

The Lock and Unlock options allow you to lock the gateway so that it does not accept any new incoming calls. This is beneficial when, for example, you are uploading new software files to the gateway and you don't want any traffic to interfere with the process.

➢ To lock the gateway, take these 4 steps:

1. Open the 'Maintenance Actions' screen (**Maintenance** menu); the 'Maintenance Actions' screen is displayed.

Figure 5-71: Maintenance Actions Screen



Maintenance Actions

RESET

Reset Board: Reset

Burn To FLASH: Yes

Graceful Option: No

LOCK / UNLOCK

Lock: LOCK

Graceful Option: Yes

Lock Timeout [sec]: -1

Current Admin State: UNLOCKED

Save Configuration

Save Configuration: BURN

2. Under the LOCK / UNLOCK group, from the 'Graceful Option' drop-down list, select one of the following options:
   - 'Yes': The gateway is 'locked' only after the user-defined time in the 'Lock Timeout' field (refer to Step 3) expires or no more active traffic exists (the earliest thereof). In addition, no new traffic is accepted.
   - 'No': The gateway is "locked" regardless of traffic. Any existing traffic is terminated immediately.
3. In the 'Lock Timeout' field (relevant only if the 'Graceful Option' in the previous step is set to 'Yes'), enter the time (in seconds) after which the gateway locks. Note that if no traffic exists and the time has not expired, the gateway locks.
4. Click the **LOCK** button; If 'Graceful Option' is set to 'Yes', the lock is delayed and a screen displaying the number of remaining calls and time is displayed. Otherwise, the lock process begins immediately. The Current Admin State displays the current state: LOCKED or UNLOCKED.

➢ To unlock the gateway, take these 2 steps:

1. Access the 'Maintenance Actions' screen as described above in the previous procedure.
2. Click the **UNLOCK** button. Unlock starts immediately and the gateway is ready for new incoming calls.

5.9.2 Saving Configuration

The 'Maintenance Actions' screen enables you to save the current parameter configuration and the loaded auxiliary files to the *non-volatile* memory (i.e., flash) so they are available after a hardware reset (or power fail). Parameters that are only saved to the *volatile* memory (RAM) revert to their previous settings after hardware reset.

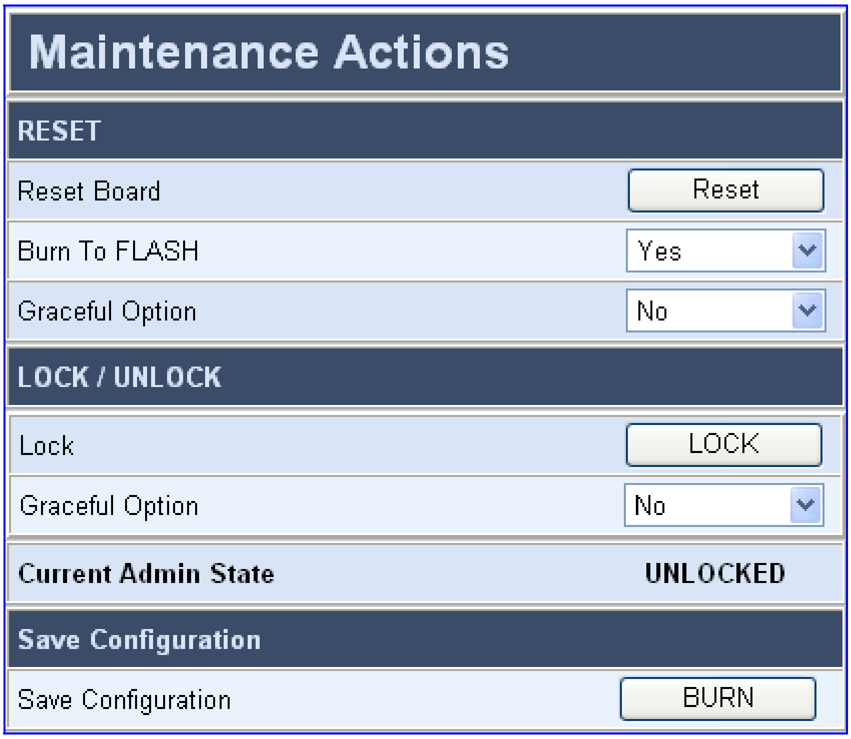
Notes:

- Saving changes to the *non-volatile* memory may disrupt traffic on the gateway. To avoid this, disable all new traffic before saving by performing a graceful lock (refer to Section 5.9.1 on page 198).
- In the Web interface, parameters prefixed with an exclamation mark ('!') are saved to the non-volatile memory only after a device reset.

➢ To save the changes to the *non-volatile*, take these 2 steps:

1. Open the 'Maintenance Actions' screen (**Maintenance** menu); the 'Maintenance Actions' screen is displayed.

Figure 5-72: Maintenance Actions Screen



Maintenance Actions

RESET

Reset Board	Reset
Burn To FLASH	Yes
Graceful Option	No

LOCK / UNLOCK

Lock	LOCK
Graceful Option	No
Current Admin State	**UNLOCKED**

Save Configuration

Save Configuration	BURN

2. Click the **BURN** button; a confirmation message appears when the save is completed successfully.

5.9.3 Resetting the Mediant 1000

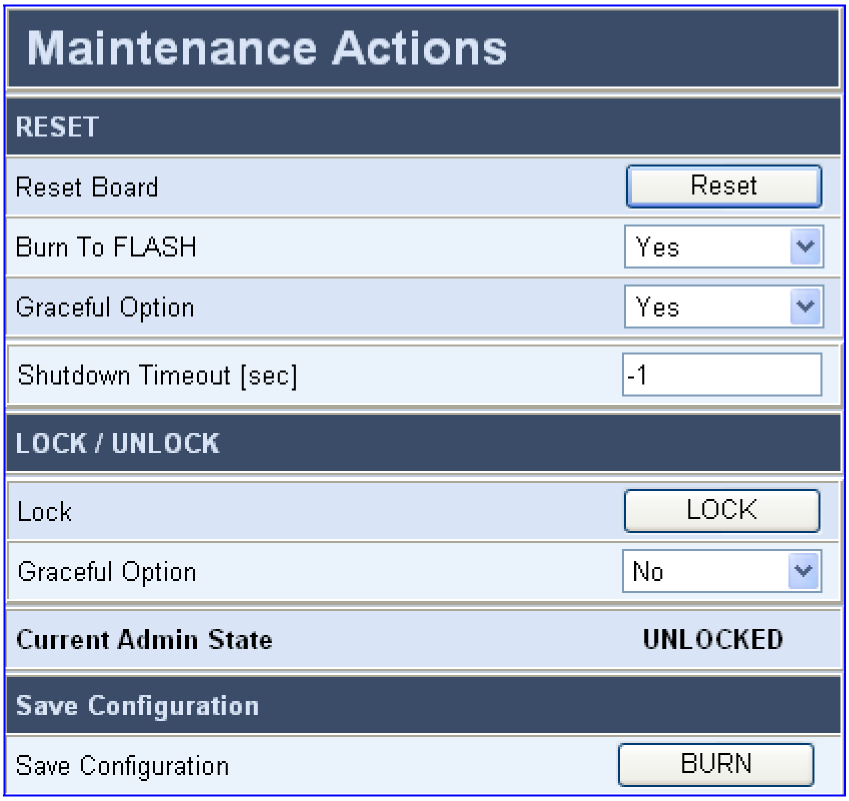
The 'Maintenance Actions' screen enables you to remotely reset the gateway. Before you reset the gateway, you can choose the following options:

- Save the gateway's current configuration to the flash memory (non-volatile).
- Perform a graceful shutdown. Reset starts only after a user-defined time expires or no more active traffic exists (the earliest thereof).

➢ To reset the gateway, take these 6 steps:

1. Open the ‘Maintenance Actions' screen (**Maintenance** menu); the ‘Maintenance Actions' screen is displayed.

Figure 5-73: Maintenance Actions Screen



Maintenance Actions

RESET

Reset Board: Reset

Burn To FLASH: Yes

Graceful Option: Yes

Shutdown Timeout [sec]: -1

LOCK / UNLOCK

Lock: LOCK

Graceful Option: No

Current Admin State: UNLOCKED

Save Configuration

Save Configuration: BURN

2. Under the RESET group, from the 'Burn To FLASH' drop-down list, select one of the following options:
   - 'Yes': The gateway's current configuration is burned (i.e., saved) to the flash memory prior to reset (default).
   - 'No': Resets the device without burning (i.e., saving) the current configuration to flash (discards all unsaved modifications to the configuration).

4. Under the RESET group, from the 'Graceful Option' drop-down list, select one of the following options:
   - 'Yes': Reset starts only after the user-defined time in the Shutdown Timeout field (refer to Step 5) expires or no more active traffic exists (the earliest thereof). In addition, no new traffic is accepted.
   - 'No': Reset starts regardless of traffic and any existing traffic is terminated at once.

5. In the 'Shutdown Timeout' field (relevant only if the 'Graceful Option' in the previous step is set to 'Yes'), enter the time after which the gateway resets. Note that if no traffic exists and the time has not expired, the gateway resets.

6. Click the **RESET** button; If 'Graceful Option' is set to 'Yes', the reset is delayed and a screen displaying the number of remaining calls and time is displayed. When the device resets, a message is displayed informing of the waiting period.

5.10 Monitoring the Gateway (Home Page)

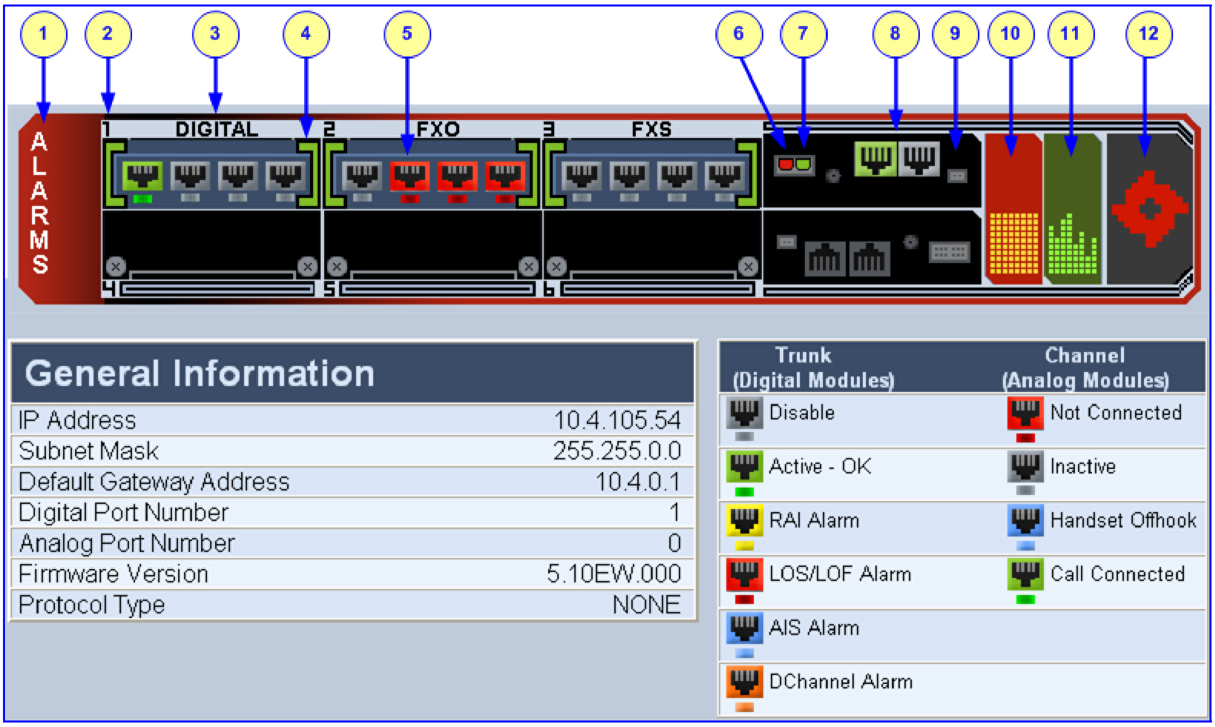
The Home icon, located above the main menu bar, opens the Home page. This page allows you to perform the following:

- Monitor trunks and channels (refer to Section 5.10.1 on page 204)
- Monitor analog and digital modules (refer to Section 5.10.2 on page 206)
- Monitor Ethernet ports, Dry Contacts, power supply units, and fan tray unit (refer to Section 5.10.3 on page 207)
- View the Active Alarms table (refer to Section 5.10.4 on page 207)
- View trunk settings (refer to Section 5.10.5 on page 208)
- View Ethernet port settings (refer to Section 5.10.6 on page 209)
- Assign a name to ports (refer to Section 5.10.7 on page 209)
- Release analog channels (refer to Section 5.10.8 on page 210)
- Replace modules (refer to Section 5.10.9 on page 210)

➢ To access the Home page, take this step:

- Open the Home page by clicking the Home icon ; the Home page is displayed.

Figure 5-74: Areas of the Home Page



General Information	
IP Address	10.4.105.54
Subnet Mask	255.255.0.0
Default Gateway Address	10.4.0.1
Digital Port Number	1
Analog Port Number	0
Firmware Version	5.10EW.000
Protocol Type	NONE

Trunk (Digital Modules)	Channel (Analog Modules)
Disable	Not Connected
Active - OK	Inactive
RAI Alarm	Handset Offhook
LOS/LOF Alarm	Call Connected
AIS Alarm	
DChannel Alarm	

The Home page also displays general information in the General Information pane. This information includes parameters such as the gateway's IP address, the number of digital and analog ports, and firmware version.

Table 5-45 on page 203 describes the areas of the graphic display of the Mediant 1000 chassis.

Table 5-45: Description of the Areas of the Home Page

Item#	Description
1	**ALARMS** button for viewing the Active Alarms table. For a detailed description, refer to Section 5.10.4 on page 207.
2	Module slot number (1 to 6).
3	Module type (digital, FXO, or FXS).
4	Module status indicator. For a detailed description, refer to Section 5.10.2 on page 206.
5	Module's port (trunk or channel) status indicator. For a detailed description, refer to Section 5.10.1 on page 204.
6	Dry Contact (normally open) status indicator. For a detailed description, refer to Section 5.10.3 on page 207.
7	Dry Contact (normally closed) status indicator. For a detailed description, refer to Section 5.10.3 on page 207.
8	Ethernet port status indicator (refer to Section 5.10.3 on page 207). If clicked, the 'Ethernet Port Information' screen opens (refer to Section 5.10.6 on page 209.
9	CPU module.
10	Power Supply Unit 1 status indicator. For a detailed description, refer to Section 5.10.3 on page 207.
11	Power Supply Unit 2 status indicator. For a detailed description, refer to Section 5.10.3 on page 207.
12	Fan tray unit status indicator. For a detailed description, refer to Section 5.10.3 on page 207.

5.10.1 Monitoring the Mediant 1000 Trunks and Channels

The Home page provides real-time monitoring of the trunks and channels.

➢ **To monitor the status of the Mediant 1000 trunks and channel ports, take this step:**

- Open the Home page by clicking the Home icon; the Home page is displayed.

The number of trunks and channels that appear in the screen depends on the system configuration. The Home page in Figure 5-74 on page 202 depicts a system with one T1 span, an FXO module with three channels, and an FXS module.

The color of each trunk and FXO/FXS channel icon indicates the status of that trunk or channel. Table 5-46 below describes the color-coding of the trunk and channel icons.

Table 5-46: Trunk and FXO/FXS Channel Status Color Indicators

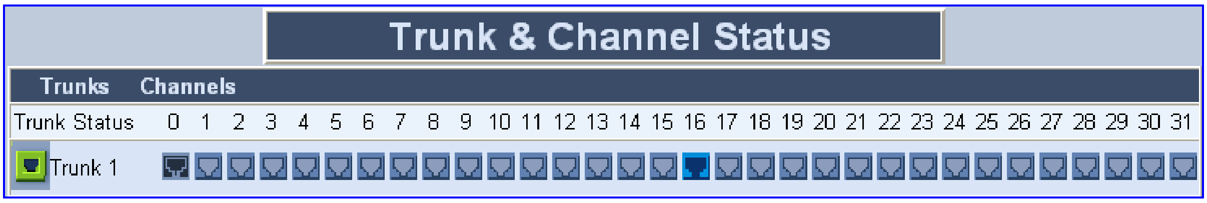
Trunk/Channel Status Icon		Trunk (Digital Module)		Channel (Analog Module)	
Indicator	**Color**	**Label**	**Description**	**Label**	**Description**
⬜	Grey	**Disable**	Trunk not configured (not in use)	**Inactive**	Channel is currently onhook
🟩	Green	**Active - OK**	Trunk synchronized	**Call Connected**	Active RTP stream
🟨	Yellow	**RAI Alarm**	Remote Alarm Indication (RAI), also known as the Yellow Alarm	--	--
🟥	Red	**LOS/LOF Alarm**	Loss due to LOS (Loss of Signal) or LOF (Loss of Frame)	**Not Connected**	No analog line is connected to this port (FXO only)
🟦	Blue	**AIS Alarm**	Alarm Indication Signal (AIS), also known as the Blue Alarm	**Handset Offhook**	Channel is offhook, but there is no active RTP session
🟧	Orange	**D Channel Alarm**	D-channel alarm	--	--

You can drill-down to view a detailed status of each channel pertaining to a trunk or FXO/FXS port.

➢ **To view a detailed status of a channel, take these 3 steps:**

1. In the Home page, click the trunk or analog port of whose status you want to view; a shortcut menu appears.
2. From the shortcut menu, choose **Port Settings**; the following appears, depending on whether you clicked a trunk or FXO/FXS port:
   - **Trunk:** the 'Trunk & Channel Status' screen pertaining to the specific trunk appears.

Figure 5-75: Status of Channels per Trunk in the Trunk & Channel Screen



Trunk & Channel Status

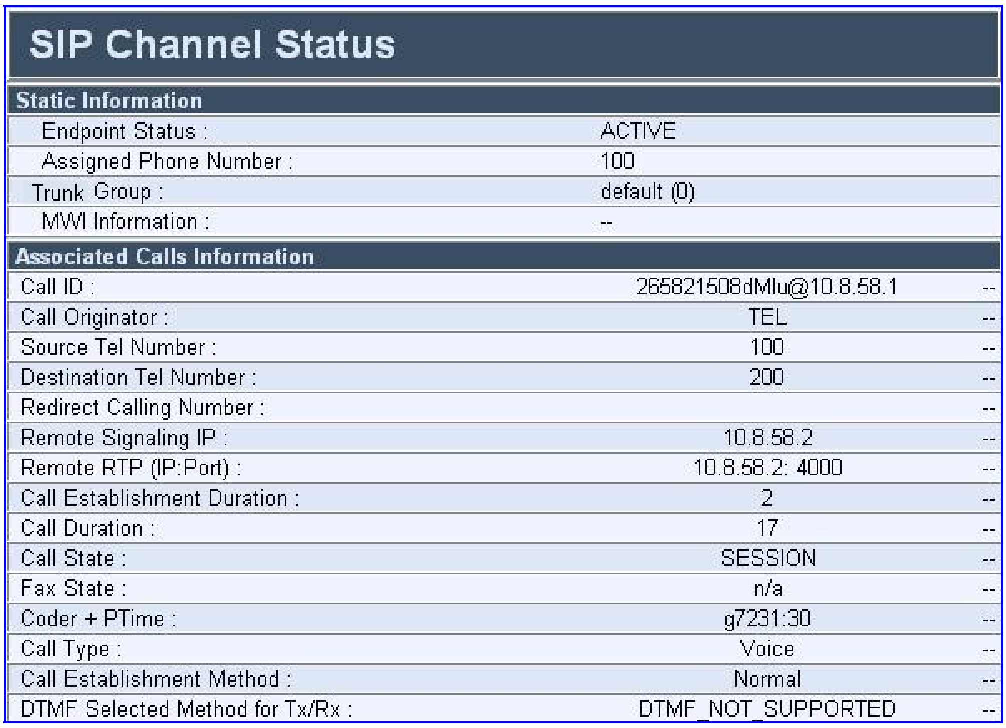
Trunks	Channels
Trunk Status	0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Trunk 1	

The trunk's channels are graphically displayed as icons. The colors of the icons depict the channels' statuses. For a description of the color coding for the channel status, refer to Table 5-47 on page 205. Click the required channel of the trunk; the channel status details screens appear (refer to Figure 5-76).

- **FXO/FXS:** the channel status details screens appear (refer to Figure 5-76).

3. Click the submenu links to view the various channel parameter statistics and settings.

Figure 5-76: Channel Status Details Screen (e.g., Trunk Channel)



SIP Channel Status

Static Information	
Endpoint Status :	ACTIVE
Assigned Phone Number :	100
Trunk Group :	default (0)
MWI Information :	--

Associated Calls Information		
Call ID :	265821508dMlu@10.8.58.1	--
Call Originator :	TEL	--
Source Tel Number :	100	--
Destination Tel Number :	200	--
Redirect Calling Number :		--
Remote Signaling IP :	10.8.58.2	--
Remote RTP (IP:Port) :	10.8.58.2: 4000	--
Call Establishment Duration :	2	--
Call Duration :	17	--
Call State :	SESSION	--
Fax State :	n/a	--
Coder + PTime :	g7231:30	--
Call Type :	Voice	--
Call Establishment Method :	Normal	--
DTMF Selected Method for Tx/Rx :	DTMF_NOT_SUPPORTED	--

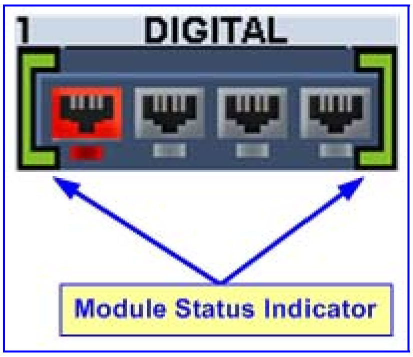
Table 5-47: Trunk Channel Status Color Indicators continues on pages 205 to 205)

Indicator	Color	Label	Description
◻	Grey	**Inactive**	Configured, but currently no call
🟩	Green	**Active**	Call in progress (RTP traffic)
🟪	Pink	**SS7**	Configured for SS7 (Currently not supported)
⬛	Dark blue	**Non Voice**	Not configured
🟦	Blue	**ISDN Signaling**	Configured as a D-channel

5.10.2 Monitoring the Modules

The Home page also provides color-coding for displaying the status of the modules (digital and anlog). In the Home page, the color of the 'square brackets' enclosing the module depicts the status of the module.

Figure 5-77: Module Status Indicators



The color coding of the module status indicators are described in the table below:

Table 5-48: Description of the Module Status Indicators

Indicator	Color	Description
[	Green	Module has been inserted or is correctly configured.
[	Grey	Module was removed. 'Reserved' is displayed alongside the module's name.
[	Red	Module failure. 'Failure' is displayed instead of the module's name.

5.10.3 Monitoring Ethernet Ports, Dry Contacts, Power Supply Units, and Fan Tray Unit

The Home page also displays the status of the Ethernet ports, Dry Contacts, power supply units, and fan tray unit. The table below describes the color-coding of the status indicators of these units:

Table 5-49: Description of Ethernet Ports, Dry Contacts, Power Supply, and Fan Tray Indicators

Item# (refer to Figure 5-74 on page 202)	Unit	Color	Description
6	Dry Contact	**Green**	Dry Contact is open (normal)
		Red	Dry contact is closed
7	Dry Contact	**Green**	Dry Contact is closed (normal)
		Red	Dry contact is open
8	Ethernet Port	**Green**	Ethernet link is working
		Grey	Ethernet link not configured
10 & 11	Power Supply Units	**Green**	Power supply is operating
		Red	Power supply failure or no power supply unit installed
12	Fan Tray Unit	**Green**	Fan tray operating
		Red	Fan tray failure

5.10.4 Viewing the Active Alarms Table

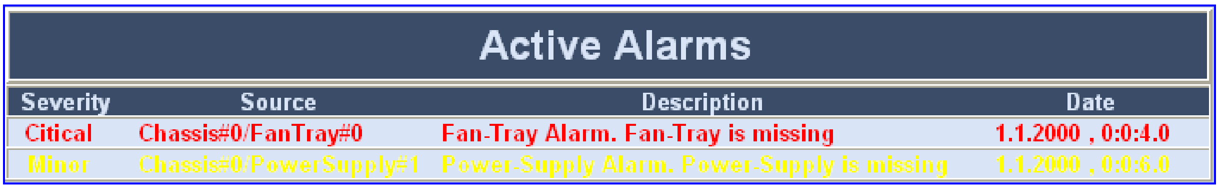
The Home page allows you to view a list of active alarms. These alarms are displayed in the Active Alarms table. For each alarm, the following is displayed:

- **Severity:** severity level of the alarm (Critical, Major, Minor)
- **Source:** module or unit from which the alarm was raised
- **Description:** brief explanation of the alarm
- **Date:** date and time that the alarm was generated

➢ To view a list of alarms, take these 2 steps:

1. Open the Home page by clicking the **Home** icon; the Home page is displayed.
2. On the graphical display of the Mediant 1000 front panel, click the the area labelled '**ALARMS**' or any area that displays the tooltip 'Click To Get Active Alarms Table'; the Active Alarms table appears.

Figure 5-78: Active Alarms Table Screen



Active Alarms

Severity	Source	Description	Date
Citical	Chassis#0/FanTray#0	Fan-Tray Alarm. Fan-Tray is missing	1.1.2000 , 0:0:4.0
Minor	Chassis#0/PowerSupply#1	Power-Supply Alarm. Power-Supply is missing	1.1.2000 , 0:0:6.0

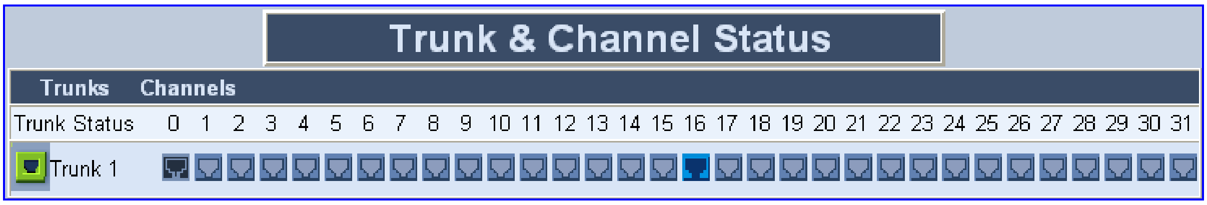
5.10.5 Viewing Trunk Settings

The Home page provides an alternative path (**Advanced Configuration** menu > **Trunk Settings**) for accessing the 'Trunk Settings' screen (refer to Section 5.6.3 on page 160), in which the trunk settings are viewed and configured.

➢ To view the trunk settings, take these 3 steps:

1. In the Home page, click the required trunk port; a shortcut menu appears.
2. From the shortcut menu, choose **Port Settings**; the 'Trunk & Channel Status' screen pertaining to the specific trunk appears.

Figure 5-79: Trunk and Channel Status Screen



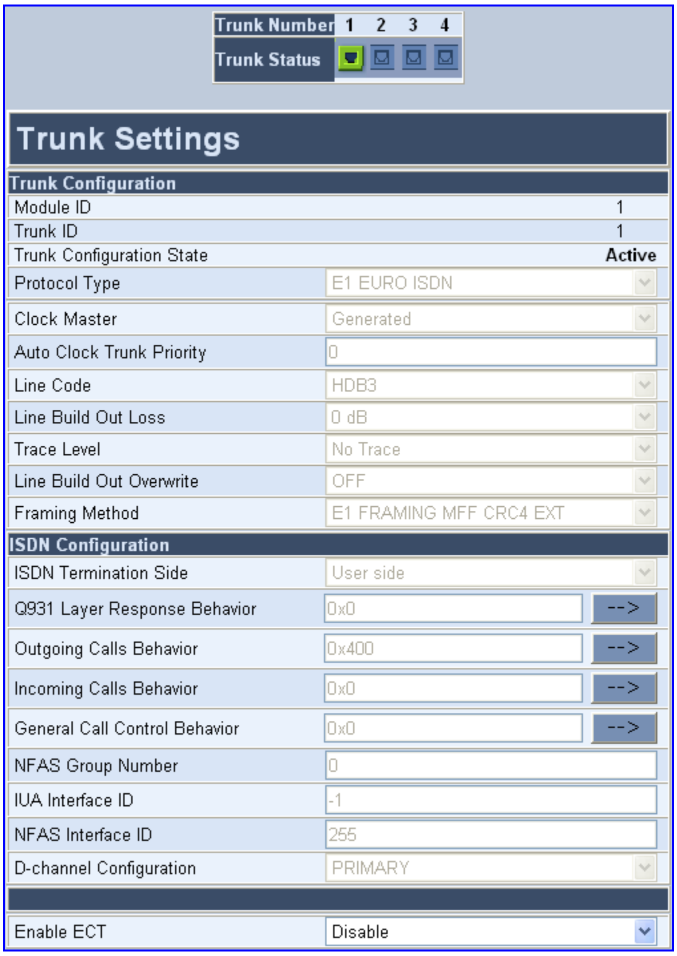
Trunk & Channel Status

Trunks Channels

Trunk Status	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Trunk 1																																

3. Click the **Trunk** icon, and then from the shortcut menu, choose **Port Settings**; the 'Trunk Settings' screen appears.

Figure 5-80: Trunk Settings Screen



Trunk Number	1	2	3	4
Trunk Status				

Trunk Settings

Trunk Configuration	
Module ID	1
Trunk ID	1
Trunk Configuration State	Active
Protocol Type	E1 EURO ISDN
Clock Master	Generated
Auto Clock Trunk Priority	0
Line Code	HDB3
Line Build Out Loss	0 dB
Trace Level	No Trace
Line Build Out Overwrite	OFF
Framing Method	E1 FRAMING MFF CRC4 EXT
ISDN Configuration	
ISDN Termination Side	User side
Q931 Layer Response Behavior	0x0 -->
Outgoing Calls Behavior	0x400 -->
Incoming Calls Behavior	0x0 -->
General Call Control Behavior	0x0 -->
NFAS Group Number	0
IUA Interface ID	-1
NFAS Interface ID	255
D-channel Configuration	PRIMARY
Enable ECT	Disable

For detailed information on configuring the 'Trunk Settings' screen, refer to Section 5.6.3 on page 160.

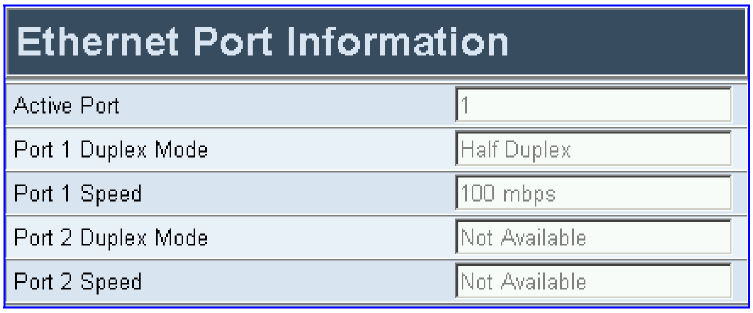
5.10.6 Viewing Ethernet Port Settings

The 'Ethernet Port Information' screen provides read-only information on the Ethernet connection used by the Mediant 1000. The Ethernet Port Information parameters are displayed in Table 5-50. For detailed information on the Ethernet interface configuration, refer to Section 10.1 on page 301.

➢ To view Ethernet port settings, take this step:

- In the Home page, click the Ethernet port status icon ; the 'Ethernet Port Information' screen opens.

Figure 5-81: Ethernet Port Information Screen



Ethernet Port Information

Active Port	1
Port 1 Duplex Mode	Half Duplex
Port 1 Speed	100 mbps
Port 2 Duplex Mode	Not Available
Port 2 Speed	Not Available

Table 5-50: Ethernet Port Information Parameters

Parameter	Description
Active Port	Shows the active Ethernet port (1 or 2).
Port 1 Duplex Mode	Shows the Duplex mode Ethernet port 1 is using (Half Duplex or Full Duplex).
Port 1 Speed	Shows the speed, in Mbps, that Ethernet port 1 is using (10 Mbps or 100 Mbps).
Port 2 Duplex Mode	Shows the Duplex mode Ethernet port 2 is using (Half Duplex or Full Duplex).
Port 2 Speed	Shows the speed, in Mbps, that Ethernet port 2 is using (10 Mbps or 100 Mbps).

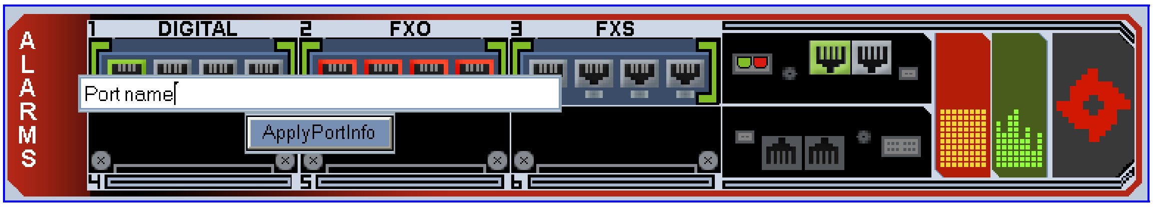
5.10.7 Assigning a Name or Brief Description to a Port

The Home page allows you to assign an arbitrary name or brief description to the gateway's ports. This description appears as a tooltip when you move your mouse over the specific port.

➢ To add a port description, take these 4 steps:

1. Open the Home page by clicking the Home icon.
2. Click the required module port icon; a shortcut menu appears.
3. From the shortcut menu, choose **Update Port Info**; a text box appears.
4. Type a brief description for the port, and then click **Apply Port Info**.

Figure 5-82: Entering a Port Name



Port name

5.10.8 Releasing an Analog Channel

The Home page allows you to inactivate (*release*) an FXO or FXS analog channel. This is sometimes useful in scenarios, for example, when the gateway (FXO) is connected to a PBX and the communication between the two can't be disconnected (e.g., when using reverse polarity).

➢ **To release a channel, take these 2 steps:**

1. Open the Home page by clicking the Home icon.
2. Click the required FXS or FXO port, and then from the shortcut menu, choose **Release Channel**; the channel is changed to inactive.

5.10.9 Replacing Modules

To replace modules (i.e., digital, FXO, and FXS), you must use the gateway's embedded Web server in combination with the physical removal and insertion of the modules. When you replace a module, you first need to 'remove' it in the Home page, then extract it physically from the chassis and physically insert a new module, and then 'insert' it in the Home page.

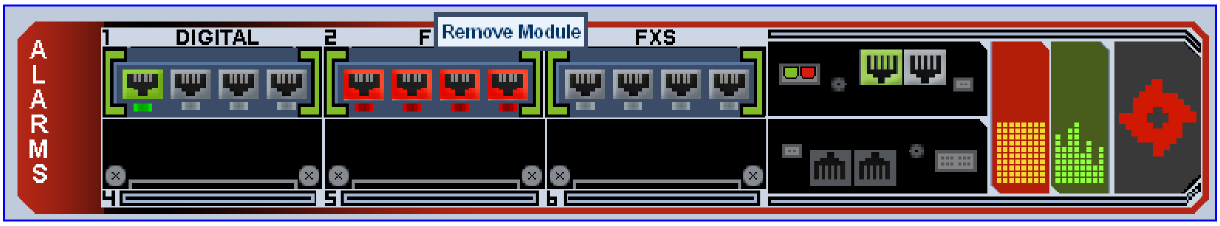
Warnings:

- Replacing of a damaged module can be performed only with the same module and in the exact module slot (e.g., a module with two digital spans in Slot 1 must be replaced with a module with two digital spans in Slot 1).
- When only one module is available, removal of the module causes the device to reset.
- Adding a module to a previously empty slot must only be performed when the power to the gateway is switched off (refer to Section 3.5.2 on page 44).

➢ **To replace a module, take these 2 steps:**

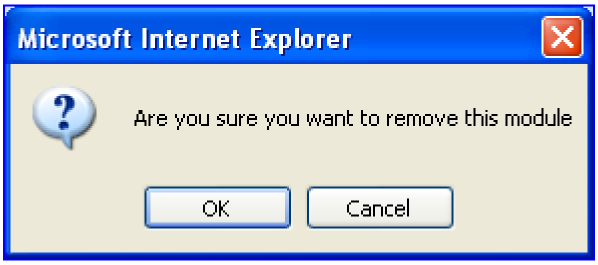
1. Remove the module by performing the following:
   a. On the Home page, click the top border line pertaining to the module that you want to replace; the **Remove Module** button appears.

Figure 5-83: Remove Module Button Appears when Clicking Module Name



   b. Click the **Remove Module** button; a message box appears requesting you to confirm module removal.

Figure 5-84: Confirmation Message Box after Clicking Remove Module Button

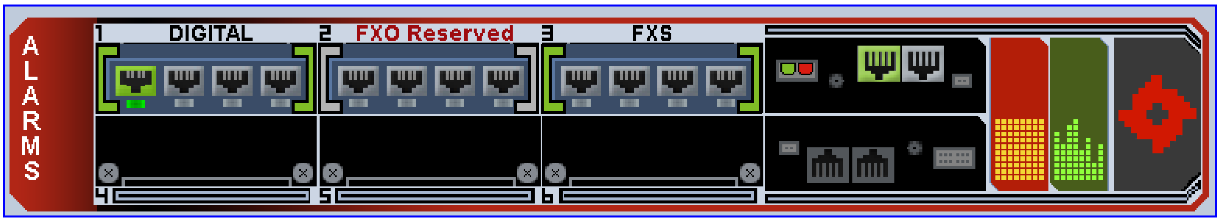


Microsoft Internet Explorer

Are you sure you want to remove this module

c. Click **OK** to confirm module removal; after a few seconds, the module is "removed" and the module status indicator is grayed. The name of the module is suffixed with the word 'Reserved'.

Figure 5-85: Module Removed Indicated by "Reserved"



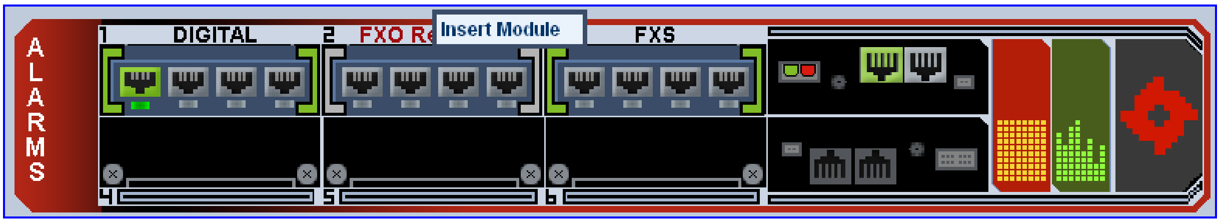
d. You can now physically remove the module (refer to Section 3.5.1 on page 44).

2. Insert the replaced module by performing the following:

e. Physically insert the replaced module (refer to Section 3.5.1 on page 44).

a. On the Home page, click the top border line pertaining to the module that you want to replace; the **Insert Module** button appears.

Figure 5-86: Insert Module Button Appears when Clicking Module Name



b. Click the **Insert Module** button; a message appears informing you that this may take a few seconds.When the message disappears, the module is inserted indicated by the disappearance of the 'Reserved' word from the module's name.

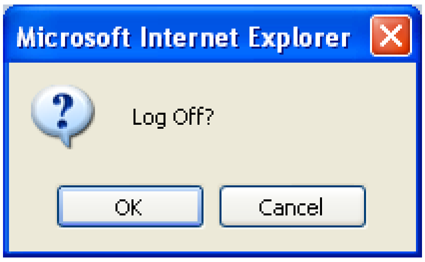
5.11 Logging Off the Embedded Web Server

The **Log Off** button enables you to log off the Embedded Web Server and to re-access it with a different account. For detailed information on the Web User Accounts, refer to Section 5.2.1 on page 55.

➢ To log off the Embedded Web Server, take these 2 steps:

1. Click the **Log Off** button on the main menu bar; the 'Log Off' prompt screen is displayed.

Figure 5-87: Log off Prompt



2. Click **OK**; the Web session is logged off.

6 *ini* File Configuration of the Mediant 1000

As an alternative to configuring the VoIP gateway using the Web interface (refer to Section 5 on page 55), it can be configured by loading the *ini* file containing Customer-configured parameters.

The *ini* file is loaded via the BootP/TFTP utility (refer to Appendix D on page 413) or via any standard TFTP server. It can also be loaded through the Web Interface (refer to Section 5.6.5 on page 168).

The *ini* file configuration parameters are stored in the Mediant 1000 non-volatile memory after the file is loaded. When a parameter is missing from the *ini* file, a default value is assigned to that parameter (according to the *cmp* file loaded on the Mediant 1000) and stored in the non-volatile memory (thereby overriding the value previously defined for that parameter). Therefore, to restore the default configuration parameters, use the *ini* file without any valid parameters or with a semicolon (;) preceding all lines in the file.

Some of the Mediant 1000 parameters are configurable through the *ini* file only (and not via the Web). These parameters usually determine a low-level functionality and are seldom changed for a specific application. For a list of the *ini* file parameters, refer to Section 6.5 on page 215.

Note: For a list of the *ini* file parameters, refer to Section 6.5 on page 215. The *ini* file parameters that are configurable through the Web, are also described in Section 5 on page 55.

6.1 Secured *ini* File

The *ini* file contains sensitive information that is required for the functioning of the Mediant 1000. It is loaded to, or retrieved from, the device via TFTP or HTTP. These protocols are unsecured and vulnerable to potential hackers. Therefore an encoded *ini* file significantly reduces these threats.

You can choose to load an encoded *ini* file to the Mediant 1000. When you load an encoded *ini* file, the retrieved *ini* file is also encoded. Use the 'TrunkPack Downloadable Conversion Utility' to encode or decode the *ini* file before you load it to, or retrieve it from the device. Note that the encoded *ini* file's loading procedure is identical to the regular *ini* file's loading procedure. For information on encoding / decoding an *ini* file, refer to Section G.1.2 on page 431.

6.2 Modifying an *ini* File

➢ To modify an *ini* file, take these 3 steps:

1. Save the *ini* file from the gateway to your PC using the Embedded Web Server (refer to Section 5.6.5 on page 168).
2. Open the file (the file is open in Notepad or a Customer-defined text file editor) and modify the *ini* file parameters according to your requirements; save and close the file.
3. Load the modified *ini* file to the gateway (using either BootP/TFTP utility or the Embedded Web Server).

This method preserves the configuration that already exists in the device, including special default values that were preconfigured when the unit was manufactured.

Tip: Before loading the *ini* file to the gateway, verify that the extension of the *ini* file saved on your PC is correct: Verify that the check box 'Hide file extension for known file types' (My computer>Tools>Folder Options>View) is unchecked. Then, confirm that the *ini* file name extension is xxx.ini and NOT erroneously xxx.ini.ini or xxx~.ini.

6.3 The *ini* File Content

The *ini* file contains the following gateway information:

- Networking parameters (refer to Section 6.5.1 on page 215)
- System parameters (refer to Section 6.5.2 on page 220)
- Web and Telnet parameters (refer to Section 6.5.3 on page 225)
- Security parameters (refer to Section 6.5.4 on page 227)
- RADIUS parameters (refer to Section 6.5.5 on page 228)
- SNMP parameters (refer to Section 6.5.6 on page 229)
- SIP Configuration parameters (refer to Section 6.5.7 on page 230)
- Voice Mail parameters (refer to Section 6.5.8 on page 237)
- ISDN and CAS Interworking-Related Parameters (refer to Section 6.5.9 on page 237)
- Analog Telephony parameters (refer to Section 6.5.10 on page 242)
- Number Manipulation and Routing parameters (refer to Section 6.5.11 on page 247)
- E1/T1 Configuration Parameters (refer to Section 6.5.12 on page 253)
- Channel Parameters (refer to Section 6.5.13 on page 255)
- Configuration Files parameters (refer to Section 6.5.14 on page 258)

6.4 The *ini* File Structure

The *ini* file can contain any number of parameters. The parameters are divided into groups by their functionality. The general form of the *ini* file is shown in Figure 6-1 below.

Figure 6-1: *ini* File Structure

```
[Sub Section Name]

Parameter_Name = Parameter_Value
Parameter_Name = Parameter_Value

; REMARK

[Sub Section Name]
```

6.4.1 The *ini* File Structure Rules

- The *ini* file name mustn’t include hyphens or spaces, use underscore instead.
- Lines beginning with a semi-colon ‘;’ (as the first character) are ignored.
- A Carriage Return must be the final character of each line.
- The number of spaces before and after ‘=’ is not relevant.
- If there is a syntax error in the parameter name, the value is ignored.
- Syntax errors in the parameter value field can cause unexpected errors (because parameters may be set to the wrong values).
- Sub-section names are optional.
- String parameters, representing file names, for example CallProgressTonesFileName, must be placed between two inverted commas (‘…’).
- The parameter name is NOT case-sensitive; the parameter value is not case-sensitive *except for coder names*.
- The *ini* file should be ended with one or more carriage returns.

6.4.2 The *ini* File Example

Figure 6-2 shows an example of an *ini* file for the VoIP gateway.

Figure 6-2: SIP *ini* File Example

```
[Channel Params]
DJBufMinDelay = 75
RTPRedundancyDepth = 1
DefaultNumber = 101
MaxDigits = 3
CoderName = g7231,90
; Phone of each endpoint
EnableSyslog = 0
[Files]
CallProgressTonesFilename = 'CPUSA.dat'
FXSLoopCharacteristicsFileName = 'coeff.dat'
SaveConfiguration = 1
```

6.5 The *ini* File Parameter Reference

The table below lists all the *ini* file parameters. References to their descriptions in the Web interface are provided, except for those *ini* file parameters that don't have corresponding Web interface parameters.

6.5.1 Networking Parameters

Table 6-1: Networking Parameters (continues on pages 215 to 220)

ini File Field Name Web Parameter Name	Valid Range and Description
EthernetPhyConfiguration	0 = 10 Base-T half-duplex 1 = 10 Base-T full-duplex 2 = 100 Base-TX half-duplex 3 = 100 Base-TX full-duplex 4 = auto-negotiate (default) For detailed information on Ethernet interface configuration, refer to Section 10.1 on page 301.
MIIRedundancyEnable	Enables the Ethernet Interface Redundancy feature. When enabled, the gateway performs a switchover to the secondary (redundant) Ethernet port upon sensing a link failure in the primary Ethernet port. When disabled, the gateway operates with a single port (i.e. no redundancy support). 0 = Disable (default) 1 = Enable For detailed information on Ethernet interface redundancy, refer to Section 10.2 on page 301. **Note:** For this parameter to take effect, a gateway reset is required.
DHCPEnable	For a description of this parameter, refer to Table 5-23 on page 139.
EnableLANWatchDog	For a description of this parameter, refer to Table 5-7 on page 85.
DNSPriServerIP	For a description of this parameter, refer to Table 5-23 on page 139.
DNSSecServerIP	For a description of this parameter, refer to Table 5-23 on page 139.

Table 6-1: Networking Parameters (continues on pages 215 to 220)

ini File Field Name Web Parameter Name	Valid Range and Description
DNS2IP	The Internal DNS table is used to resolve host names to IP addresses. Two different IP addresses (in dotted format notation) can be assigned to a hostname. The format is as follows: DNS2IP = <Hostname>, <first IP address>, <second IP address> **Note 1:** If the internal DNS table is configured, the gateway first tries to resolve a domain name using this table. If the domain name isn't found, the gateway performs a DNS resolution using an external DNS server. **Note 2:** This parameter can appear up to 10 times.
SRV2IP	Defines the Internal SRV table used for resolving host names to DNS A-Records. Three different A-Records can be assigned to a hostname. Each A-Record contains the host name, priority, weight, and port. SRV2IP = <Internal Domain Name>, <Transport Type>, <DNS Name 1>, <Priority 1>, <Weight 1>, <Port 1>, <DNS Name 2>, <Priority 2>, <Weight 2>, <Port 2>, <DNS Name 3>, <Priority 3>, <Weight 3>, <Port 3> **Note 1**: If the internal SRV table is configured, the gateway first tries to resolve a domain name using this table. If the domain name isn't found, the gateway performs an SRV resolution using an external DNS server. **Note 2**: This parameter can appear up to 10 times. **Note 3:** To configure the Internal SRV table using the Web interface, refer to Section 5.5.4.5 on page 112.
EnableSTUN	For a description of this parameter, refer to Table 5-24 on page 142.
STUNServerPrimaryIP	For a description of this parameter, refer to Table 5-24 on page 142.
STUNServerSecondaryIP	For a description of this parameter, refer to Table 5-24 on page 142.
STUNServerDomainName	Defines the domain name for the Simple Traversal of User Datagram Protocol (STUN) server's address (used for retrieving all STUN servers with an SRV query). The STUN client can perform the required SRV query to resolve this domain name to an IP address and port, sort the server list, and use the servers according to the sorted list. **Note:** Use either the STUNServerPrimaryIP or the STUNServerDomainName parameter, with priority to the first one.
NATBindingDefaultTimeout	Defines the default NAT binding lifetime in seconds. STUN is used to refresh the binding information after this time expires. The valid range is 0 to 2,592,000. The default value is 30.
DisableNAT	Enables / disables the Network Address Translation (NAT) mechanism. 0 = Enabled. 1 = Disabled (default). **Note:** The compare operation that is performed on the IP address is enabled by default and is controlled by the parameter 'EnableIPAddrTranslation'. The compare operation that is performed on the UDP port is disabled by default and is controlled by the parameter 'EnableUDPPortTranslation'.
EnableIPAddrTranslation	0 = Disable IP address translation. 1 = Enable IP address translation for RTP, RTCP and T.38 packets (default). 2 = Enable IP address translation for ThroughPacket™. 3 = Enable IP address translation for all protocols (RTP, RTCP, T38 and ThroughPacket™). When enabled, the gateway compares the source IP address of the first incoming packet, to the remote IP address stated in the opening of the channel. If the two IP addresses don't match, the NAT mechanism is activated. Consequently, the remote IP address of the outgoing stream is replaced by the source IP address of the first incoming packet. **Note:** The NAT mechanism must be enabled for this parameter to take effect (DisableNAT = 0).

Table 6-1: Networking Parameters (continues on pages 215 to 220)

***ini* File Field Name** **Web Parameter Name**	**Valid Range and Description**
EnableUDPPortTranslation	0 = Disable UDP port translation (default). 1 = Enable UDP port translation. When enabled, the gateway compares the source UDP port of the first incoming packet, to the remote UDP port stated in the opening of the channel. If the two UDP ports don't match, the NAT mechanism is activated. Consequently, the remote UDP port of the outgoing stream is replaced by the source UDP port of the first incoming packet. **Note:** The NAT mechanism and the IP address translation must be enabled for this parameter to take effect (DisableNAT = 0, EnableIpAddrTranslation = 1).
NoOperationSendingMode	Enables or disables the transmission of RTP or T.38 No-Op packets. Valid options include: 0 = Disable (default) 1 = Enable This mechanism ensures that the NAT binding remains open during RTP or T.38 silence periods.
RTPNoOpEnable	This parameter is obsolete; use the parameter NoOperationSendingMode.
NoOpInterval	Defines the time interval in which RTP or T.38 No-Op packets are sent in the case of silence (no RTP / T.38 traffic) when No-Op packet transmission is enabled. The valid range is 20 to 65,000 msec. The default is 10,000. **Note:** To enable No-Op packet transmission, use the NoOperationSendingMode parameter.
RTPNoOpInterval	This parameter is obsolete; use the parameter NoOpInterval.
RTPNoOpPayloadType	Determines the payload type of No-Op packets. the valid range is 96 to 127. The default value is 120.
EnableDetectRemoteMACChange	Changes the RTP packets according to the MAC address of received RTP packets and according to Gratuitous Address Resolution Protocol (GARP) messages. Valid options include: 0 = nothing is changed. 1 = If the gateway receives RTP packets with a different source MAC address (than the MAC address of the transmitted RTP packets), then it sends RTP packets to this MAC address and removes this IP entry from the gateway's ARP cache table. 2 = The gateway uses the received GARP packets to change the MAC address of the transmitted RTP packets. 3 = both 1 and 2 options above are used (default).
StaticNatIP	For a description of this parameter, refer to Table 5-23 on page 139.
SyslogServerIP	For a description of this parameter, refer to Table 5-37 on page 179.
SyslogServerPort	For a description of this parameter, refer to Table 5-37 on page 179.
EnableSyslog	For a description of this parameter, refer to Table 5-37 on page 179.
BaseUDPport	For a description of this parameter, refer to Table 5-30 on page 155.
RemoteBaseUDPPort	For a description of this parameter, refer to Table 5-30 on page 155.
L1L1ComplexTxUDPPort	For a description of this parameter, refer to Table 5-30 on page 155.
L1L1ComplexRxUDPPort	For a description of this parameter, refer to Table 5-30 on page 155.
NTPServerIP	For a description of this parameter, refer to Table 5-24 on page 142.
NTPServerUTCOffset	For a description of this parameter, refer to Table 5-24 on page 142.
NTPUpdateInterval	For a description of this parameter, refer to Table 5-24 on page 142.

Table 6-1: Networking Parameters (continues on pages 215 to 220)

ini File Field Name Web Parameter Name	Valid Range and Description

IP Routing Table parameters:
The IP routing *ini* file parameters are array parameters. Each parameter configures a specific column in the IP routing table. The first entry in each parameter refers to the first row in the IP routing table, the second entry to the second row and so forth.
In the following example two rows are configured when the gateway is in network 10.31.x.x:
RoutingTableDestinationsColumn = 130.33.4.6, 83.4.87.6
RoutingTableDestinationMasksColumn = 255.255.255.255, 255.255.255.0
RoutingTableGatewaysColumn = 10.31.0.1, 10.31.0.112
RoutingTableInterfacesColumn = 0, 1
RoutingTableHopsCountColumn = 20, 20	

RoutingTableDestinationsColumn	For a description of this parameter, refer to Table 5-26 on page 146.
RoutingTableDestinationMasksColumn	For a description of this parameter, refer to Table 5-26 on page 146.
RoutingTableGatewaysColumn	For a description of this parameter, refer to Table 5-26 on page 146.
RoutingTableHopsCountColumn	For a description of this parameter, refer to Table 5-26 on page 146.
RoutingTableInterfacesColumn	For a description of this parameter, refer to Table 5-26 on page 146.
VLAN Parameters	
VLANMode	For a description of this parameter, refer to Table 5-27 on page 148.
VLANNativeVLANID	For a description of this parameter, refer to Table 5-27 on page 148.
VLANOamVLANID	For a description of this parameter, refer to Table 5-27 on page 148.
VLANControlVLANID	For a description of this parameter, refer to Table 5-27 on page 148.
VLANMediaVLANID	For a description of this parameter, refer to Table 5-27 on page 148.
VLANNetworkServiceClassPriority	For a description of this parameter, refer to Table 5-27 on page 148.
VLANPremiumServiceClassMediaPriority	For a description of this parameter, refer to Table 5-27 on page 148.
VLANPremiumServiceClassControlPriority	For a description of this parameter, refer to Table 5-27 on page 148.
VlanGoldServiceClassPriority	For a description of this parameter, refer to Table 5-27 on page 148.
VLANBronzeServiceClassPriority	For a description of this parameter, refer to Table 5-27 on page 148.
EnableDNSasOAM	This parameter applies to both Multiple IPs and VLAN mechanisms. Multiple IPs: Determines the network type for DNS services. VLAN: Determines the traffic type for DNS services. 1 = OAM (default) 0 = Control.
EnableNTPasOAM	This parameter applies to both Multiple IPs and VLAN mechanisms. Multiple IPs: Determines the network type for NTP services. VLAN: Determines the traffic type for NTP services. 1 = OAM (default) 0 = Control.
VLANSendNonTaggedOnNative	Specify whether to send non-tagged packets on the native VLAN. 0 = Sends priority tag packets (default). 1 = Sends regular packets (with no VLAN tag).
Multiple IPs Parameters	
EnableMultipleIPs	For a description of this parameter, refer to Table 5-23 on page 139.

Table 6-1: Networking Parameters (continues on pages 215 to 220)

ini File Field Name Web Parameter Name	Valid Range and Description
LocalMediaIPAddress	For a description of this parameter, refer to Table 5-23 on page 139.
LocalMediaSubnetMask	For a description of this parameter, refer to Table 5-23 on page 139.
LocalMediaDefaultGW	For a description of this parameter, refer to Table 5-23 on page 139.
LocalControlIPAddress	For a description of this parameter, refer to Table 5-23 on page 139.
LocalControlSubnetMask	For a description of this parameter, refer to Table 5-23 on page 139.
LocalControlDefaultGW	For a description of this parameter, refer to Table 5-23 on page 139.
LocalOAMIPAddress	For a description of this parameter, refer to Table 5-23 on page 139.
LocalOAMSubnetMask	For a description of this parameter, refer to Table 5-23 on page 139.
LocalOAMDefaultGW	For a description of this parameter, refer to Table 5-23 on page 139.
PPPoE Parameters	
EnablePPPoE	Enables the PPPoE (Point-to-Point Protocol over Ethernet) feature. 0 = Disable (default) 1 = Enable
PPPoEUserName	User Name for PAP or Host Name for CHAP authentication. The valid range is a string of up to 47 characters. The default value is 0.
PPPoEPassword	Password for PAP or Secret for CHAP authentication. The valid range is a string of up to 47 characters. The default value is 0.
PPPoEServerName	Server Name for CHAP authentication. The valid range is a string of up to 47 characters. The default value is 0.
PPPoEStaticIPAddress	IP address to use in a static configuration setup. If set, used during PPP negotiation to request this specific IP address from the PPP server. If approved by the server, this IP address is used during the session. The valid IP address range is in dotted notation xxx.xxx.xxx.xxx. The default value is 0.0.0.0.
PPPoERecoverIPAddress	IP address to use when booting from the flash to non-PPPoE (Point-to-Point Protocol over Ethernet) environments. The valid IP address range is in dotted notation xxx.xxx.xxx.xxx. The default value is 10.4.10.4.
PPPoERecoverSubnetMask	Subnet Mask to use when booting from the flash to non-PPPoE (Point-to-Point Protocol over Ethernet) environments. The valid IP address range is in dotted notation xxx.xxx.xxx.xxx. The default value is 255.255.0.0.
PPPoERecoverDfGWAddress	Default GW address to use when booting from the flash to non-PPPoE (Point-to-Point Protocol over Ethernet) environments. The valid IP address range is in dotted notation xxx.xxx.xxx.xxx. The default value is 10.4.10.1.
PPPoELCPEchoEnable	Enables or disables the Point-to-Point Protocol over Ethernet (PPPoE) disconnection auto-detection feature. Valid options include: 0 = Disable 1 = Enable (default) By default, the PPPoE Client (i.e., embedded in the gateway) sends LCP Echo packets to the server to check that the PPPoE connection is open. Some Access Concentrators (PPPoE servers) don't reply to these LCP Echo requests, resulting in a disconnection. By disabling the LCP disconnection auto-detection feature, the PPPoE Client doesn't send LCP Echo packets to the server (and does not detect PPPoE disconnections).

Table 6-1: Networking Parameters (continues on pages 215 to 220)

ini File Field Name Web Parameter Name	Valid Range and Description
Differential Services. For detailed information on IP QoS via Differentiated Services, refer to Section 10.9 on page 308.	
NetworkServiceClassDiffServ	For a description of this parameter, refer to Table 5-23 on page 139.
PremiumServiceClassMediaDiffServ	For a description of this parameter, refer to Table 5-23 on page 139.
PremiumServiceClassControlDiffServ	For a description of this parameter, refer to Table 5-23 on page 139.
GoldServiceClassDiffServ	For a description of this parameter, refer to Table 5-23 on page 139.
BronzeServiceClassDiffServ	For a description of this parameter, refer to Table 5-23 on page 139.
NFS Table Parameters (NFSServers). For an NFS *ini* file example, refer to Figure 5-39 on page 145.	
NFSServers_Index	For a description of this parameter, refer to Table 5-25 on page 145.
NFSServers_HostOrIP	For a description of this parameter, refer to Table 5-25 on page 145.
NFSServers_RootPath	For a description of this parameter, refer to Table 5-25 on page 145.
The combination of Host / IP and Root Path must be unique for each row in the table. For example, there must be only one row in the table with a Host / IP of 192.168.1.1 and Root Path of /audio.	
NFSServers_NfsVersion	For a description of this parameter, refer to Table 5-25 on page 145.
NFSServers_AuthType]	For a description of this parameter, refer to Table 5-25 on page 145.
NFSServers_UID	For a description of this parameter, refer to Table 5-25 on page 145.
NFSServers_GID	For a description of this parameter, refer to Table 5-25 on page 145.
NFSServers_VLANType]	For a description of this parameter, refer to Table 5-25 on page 145.

6.5.2 System Parameters

Table 6-2: System Parameters (continues on pages 220 to 225)

ini File Field Name Web Parameter Name	Valid Range and Description
GroundKeyDetection	Enables analog ground key detection (FXS and FXO modules implement ground start signaling) per gateway. When disabled, the gateway uses loop start signaling. Valid options include: 0 = Disable (default) 1= Enable (enables ground start) **Note 1:** For ground start signaling, ensure that the FXO G module is installed (and not the regular FXO module) in the Mediant 1000. **Note 2:** For ground start FXO, the following parameters should be configured: EnableCurrentDisconnect = 1; FXOBetweenRingTime = 300.

Table 6-2: System Parameters (continues on pages 220 to 225)

ini File Field Name Web Parameter Name	Valid Range and Description
EnableDiagnostics	Checks the correct functionality of the different hardware components on the gateway. On completion of the check, if the test fails, the gateway sends information on the test results of each hardware component to the Syslog server. 0 = Rapid and Enhanced self-test mode (default). 1 = Detailed self-test mode (full test of DSPs, PCM, Switch, LAN, PHY and Flash). 2 = A quicker version of the Detailed self-test mode (full test of DSPs, PCM, Switch, LAN, PHY, but partial test of Flash). For detailed information, refer to Section 15.1 on page 365.
WatchDogStatus	0 = Disable gateway's watch dog. 1 = Enable gateway's watch dog (default).
LifeLineType	The Lifeline is activated on: 0 = Power down (default) 1 = Power down or when link is down (physical disconnect) 2 = Power down or when link is down or on network failure (logical link disconnect) **Note:** To enable Lifeline switching on network failure, LAN watch dog must be activated (EnableLANWatchDog=1).
GWAppDelayTime	For a description of this parameter, refer to Table 5-7 on page 85.
ActivityListToLog	The Activity Log mechanism enables the gateway to send log messages (to a Syslog server) that report certain types of web actions according to a pre-defined filter. The following filters are available: **PVC** (Parameters Value Change) - Changes made on-the-fly to parameters. **AFL** (Auxiliary Files Loading) - Loading of auxiliary files (e.g., via Certificate screen). **DR** (Device Reset) - Device reset via the Maintenance screen. **FB** (Flash Memory Burning) - Burning of files / parameters to flash (e.g., Maintenance screen). **SWU** (Device Software Update) - cmp loading via the Software Upgrade Wizard. **ARD** (Access to Restricted Domains) - Access to Restricted Domains. The following screens are restricted: (1) ini parameters (AdminPage) (2) General Security Settings (3) Configuration File (4) IPSec/IKE tables (5) Software Upgrade Key (6) Internal Firewall (7) Web Access List. (8) Web User Accounts **NAA** (Non Authorized Access) - Attempt to access the Embedded Web Server with a false / empty username or password. **SPC** (Sensitive Parameters Value Change) - Changes made to sensitive parameters: (1) IP Address (2) Subnet Mask (3) Default Gateway IP Address (4) ActivityListToLog For example: ActivityListToLog = 'pvc', 'afl', 'dr', 'fb', 'swu', 'ard', 'naa', 'spc'

Table 6-2: System Parameters (continues on pages 220 to 225)

ini File Field Name Web Parameter Name	Valid Range and Description
ECHybridLoss	Sets the four wire to two wire worst case Hybrid loss, the ratio between the signal level sent to the hybrid and the echo level returning from the hybrid. 0 = 6 dB (default) 1 = 9 dB 2 = 0 dB 3 = 3 dB
GwDebugLevel	For a description of this parameter, refer to Table 5-7 on page 85.
CDRReportLevel	For a description of this parameter, refer to Table 5-7 on page 85.
CDRSyslogServerIP	For a description of this parameter, refer to Table 5-7 on page 85.
HeartBeatDestIP	Destination IP address (in dotted format notation) to which the gateway sends proprietary UDP 'ping' packets. The default IP address is 0.0.0.0.
HeartBeatDestPort	Destination UDP port to which the heartbeat packets are sent. The range is 0 to 64000. The default is 0.
HeartBeatIntervalmsec	Delay (in msec) between consecutive heartbeat packets. 10 = 100000. -1 = disabled (default).
EnableRAI	0 = Disable RAI (Resource Available Indication) service (default). 1 = Enable RAI service. If RAI is enabled, an SNMP 'acBoardCallResourcesAlarm' Alarm Trap is sent if gateway resources fall below a predefined (configurable) threshold.
RAIHighThreshold	High Threshold (in percentage) that defines the gateway's busy endpoints. The range is 0 to 100. The default value is 90%. When the percentage of the gateway's busy endpoints exceeds the value configured in High Threshold, the gateway sends an SNMP 'acBoardCallResourcesAlarm' Alarm Trap with a 'major' Alarm Status. **Note:** The gateway's available Resources are calculated by dividing the number of busy endpoints by the total number of available gateway endpoints.
RAILowThreshold	Low Threshold (in percentage) that defines the gateway's busy endpoints. The range is 0 to 100. The default value is 90%. When the percentage of the gateway's busy endpoints falls below the value defined in Low Threshold, the gateway sends an SNMP 'acBoardCallResourcesAlarm' Alarm Trap with a 'cleared' Alarm Status.
RAILoopTime	Time interval (in seconds) that the gateway checks for resource availability. The default is 10 seconds.
Disconnect Supervision Parameters	
DisconnectOnBrokenConnection	For a description of this parameter, refer to Table 5-7 on page 85.
BrokenConnectionEventTimeout	For a description of this parameter, refer to Table 5-7 on page 85.
EnableSilenceDisconnect	For a description of this parameter, refer to Table 5-7 on page 85.
FarEndDisconnectSilencePeriod	For a description of this parameter, refer to Table 5-7 on page 85.
FarEndDisconnectSilenceMethod	For a description of this parameter, refer to Table 5-7 on page 85.

Table 6-2: System Parameters (continues on pages 220 to 225)

ini File Field Name Web Parameter Name	Valid Range and Description
FarEndDisconnectSilenceThreshold	Threshold of the packet count (in percents), below which is considered silence by the media gateway. The valid range is 1 to 100. The default is 8%. **Note:** Applicable only if silence is detected according to packet count (FarEndDisconnectSilenceMethod = 1).
Automatic Update Parameters	
PrtFileURL	Specifies the name of the Prerecorded Tones file and the location of the server (IP address or FQDN) from which it is loaded. http://server_name/file, https://server_name/file. **Note:** The maximum length of the URL address is 99 characters.
CptFileURL	Specifies the name of the CPT file and the location of the server (IP address or FQDN) from which it is loaded. http://server_name/file, https://server_name/file. **Note:** The maximum length of the URL address is 99 characters.
CmpFileURL	Specifies the name of the *cmp* file and the location of the server (IP address or FQDN) from which the gateway loads a new *cmp* file and updates itself. The *cmp* file can be loaded using: HTTP, HTTPS, FTP, FTPS or NFS. For example: http://192.168.0.1/filename **Note 1:** When this parameter is set in the *ini* file, the gateway always loads the *cmp* file after it is reset. **Note 2:** The *cmp* file is validated before it is burned to flash. The checksum of the *cmp* file is also compared to the previously-burnt checksum to avoid unnecessary resets. **Note 3:** The maximum length of the URL address is 99 characters.
IniFileURL	Specifies the name of the *ini* file and the location of the server (IP address or FQDN) from which the gateway loads the *ini* file. The *ini* file can be loaded using: HTTP, HTTPS, FTP, FTPS or NFS. For example: http://192.168.0.1/filename http://192.8.77.13/config<MAC> https://<username>:<password>@<IP address>/<file name> **Note 1:** When using HTTP or HTTPS, the date and time of the *ini* file are validated. Only more recently-dated *ini* files are loaded. **Note 2:** The optional string '<MAC>' is replaced with the gateway's MAC address. Therefore, the gateway requests an *ini* file name that contains its MAC address. This option enables loading different configurations for specific gateways. **Note 3:** The maximum length of the URL address is 99 characters.
FXSCoeffFileURL	Specifies the name of the FXS coefficients file and the location of the server (IP address or FQDN) from which it is loaded. http://server_name/file, https://server_name/file. **Note:** The maximum length of the URL address is 99 characters.
FXOCoeffFileURL	Specifies the name of the FXO coefficients file and the location of the server (IP address or FQDN) from which it is loaded. http://server_name/file, https://server_name/file. **Note:** The maximum length of the URL address is 99 characters.
CasFileURL	Specifies the name of the CAS file and the location of the server (IP address or FQDN) from which it is loaded. http://server_name/file, https://server_name/file. **Note:** The maximum length of the URL address is 99 characters.
UserInfoFileURL	Specifies the name of the User Information file and the location of the server (IP address or FQDN) from which it is loaded. http://server_name/file, https://server_name/file. **Note:** The maximum length of the URL address is 99 characters.

Table 6-2: System Parameters (continues on pages 220 to 225)

ini File Field Name Web Parameter Name	Valid Range and Description
AutoUpdateCmpFile	Enables / disables the Automatic Update mechanism for the cmp file. 0 = The Automatic Update mechanism doesn't apply to the cmp file (default). 1 = The Automatic Update mechanism includes the cmp file.
AutoUpdateFrequency	Determines the number of minutes the gateway waits between automatic updates. The default value is 0 (the update at fixed intervals mechanism is disabled).
AutoUpdatePredefinedTime	Schedules an automatic update to a predefined time of the day. The range is 'HH:MM' (24-hour format). For example: 20:18 **Note:** The actual update time is randomized by five minutes to reduce the load on the Web servers.
ResetNow	Invokes an immediate restart of the gateway. This option can be used to activate offline (not on-the-fly) parameters that are loaded via IniFileUrl. 0 = The immediate restart mechanism is disabled (default). 1 = The gateway immediately restarts after an ini file with this parameter set to 1 is loaded.
BootP and TFTP Parameters	
The BootP parameters are special 'Hidden' parameters. Once defined and saved in the flash memory, they are used even if they don't appear in the *ini* file.	
BootPRetries	**Note:** This parameter only takes effect from the next reset of the gateway. This parameter is used to: Set the number of BootP requests the gateway sends during start-up. The gateway stops sending BootP requests when either BootP reply is received or number of retries is reached. 1 = 1 BootP retry, 1 second. 2 = 2 BootP retries, 3 second. 3 = 3 BootP retries, 6 second (default). 4 = 10 BootP retries, 30 second. 5 = 20 BootP retries, 60 second. 6 = 40 BootP retries, 120 second. 7 = 100 BootP retries, 300 second. 15 = BootP retries indefinitely. Set the number of DHCP packets the gateway sends. After all packets were sent, if there's still no reply, the gateway loads from flash. 1 = 4 DHCP packets 2 = 5 DHCP packets 3 = 6 DHCP packets (default) 4 = 7 DHCP packets 5 = 8 DHCP packets 6 = 9 DHCP packets 7 = 10 DHCP packets 15 = 18 DHCP packets
BootPSelectiveEnable	Enables the Selective BootP mechanism. 1 = Enabled. 0 = Disabled (default). The Selective BootP mechanism (available from Boot version 1.92) enables the gateway's integral BootP client to filter unsolicited BootP/DHCP replies (accepts only BootP replies that contain the text 'AUDC' in the vendor specific information field). This option is useful in environments where enterprise BootP/DHCP servers provide undesired responses to the gateway's BootP requests. **Note:** When working with DHCP (DHCPEnable = 1) the selective BootP feature must be disabled.
BootPDelay	The interval between the device's startup and the first BootP/DHCP request that is issued by the device. 1 = 1 second (default). 2 = 3 second. 3 = 6 second. 4 = 30 second. 5 = 60 second. **Note:** This parameter only takes effect from the next reset of the device.

Table 6-2: System Parameters (continues on pages 220 to 225)

ini File Field Name Web Parameter Name	Valid Range and Description
ExtBootPReqEnable	0 = Disable (default). 1 = Enable extended information to be sent in BootP request. If enabled, the device uses the vendor specific information field in the BootP request to provide device-related initial startup information such as board type, current IP address, software version, etc. For a full list of the vendor specific Information fields, refer to Section 7.3.2 on page 261. The BootP/TFTP configuration utility displays this information in the 'Client Info' column (refer to Figure D-1). **Note:** This option is not available on DHCP servers.
Serial parameters	
DisableRS232	0 = RS-232 serial port is enabled (default). 1 = RS-232 serial port is disabled. The RS-232 serial port can be used to change the networking parameters (Section 4.2.4 on page 50) and view error / notification messages. For information on establishing a serial communications link with the Mediant 1000, refer to Section 12.2 on page 319).
SerialBaudRate	Determines the value of the RS-232 baud rate. The valid range is any value. It is recommended to use the following standard values: 1200, 2400, 9600 (default), 14400, 19200, 38400, 57600, 115200.
SerialData	Determines the value of the RS-232 data bit. 7 = 7-bit. 8 = 8-bit (default).
SerialParity	Determines the value of the RS-232 polarity. 0 = None (default). 1 = Odd. 2 = Even.
SerialStop	Determines the value of the RS-232 stop bit. 1 = 1-bit (default). 2 = 2-bit.
SerialFlowControl	Determines the value of the RS-232 flow control. 0 = None (default). 1 = Hardware.

6.5.3 Web and Telnet Parameters

Table 6-3: Web and Telnet Parameters (continues on pages 225 to 226)

ini File Field Name Web Parameter Name	Valid Range and Description
WebAccessList_x	Defines up to ten IP addresses that are permitted to access the gateway's Web and Telnet interfaces. Access from an undefined IP address is denied. This security feature is inactive (the gateway can be accessed from any IP address) when the table is empty. For example: WebAccessList_0 = 10.13.2.66 WebAccessList_1 = 10.13.77.7 The default value is 0.0.0.0 (the gateway can be accessed from any IP address). For defining the Web and Telnet Access list using the emedded Web server, refer to Section 5.6.7.2 on page 172.
WebRADIUSLogin	For a description of this parameter, refer to Table 5-36 on page 176.
DisableWebTask	0 = Enable Web management (default). 1 = Disable Web management.

Table 6-3: Web and Telnet Parameters (continues on pages 225 to 226)

ini File Field Name Web Parameter Name	Valid Range and Description
ResetWebPassword	Resets the username and password of the primary and secondary accounts to their defaults. 0 = Password and username retain their values (default). 1 = Password and username are reset (for the default username and password, refer to Table 4-1 on page 47). **Note:** The username and password cannot be reset from the Web (i.e., via AdminPage or by loading an *ini* file).
DisableWebConfig	0 = Enable changing parameters from Web (default). 1 = Operate Web Server in 'read only' mode.
HTTPport	HTTP port used for Web management (default = 80)
Telnet Parameters	
TelnetServerEnable	For a description of this parameter, refer to Table 5-24 on page 142.
TelnetServerPort	For a description of this parameter, refer to Table 5-24 on page 142.
TelnetServerIdleDisconnect	For a description of this parameter, refer to Table 5-24 on page 142.
Customizing the Web Appearance Parameters For detailed information on customizing the Web Interface, refer to Section 12.6 on page 328.	
UseProductName	0 = Disabled (default). 1 = Enabled. If enabled, the 'UserProductNane' text string is displayed instead of the default product name.
UserProductName	Text string that replaces the product name. The default is 'Mediant 1000'. The string can be up to 29 characters.
UseWebLogo	0 = Logo image is used (default). 1 = Text string is used instead of a logo image. If enabled, AudioCodes' default logo (or any other logo defined by the 'LogoFileName' parameter) is replaced with a text string defined by the 'WebLogoText' parameter.
WebLogoText	Text string that replaces the logo image. The string can be up to 15 characters.
LogoWidth	Width (in pixels) of the logo image. **Note:** The optimal setting depends on the resolution settings. The default value is 441, which is the width of AudioCodes' displayed logo.
LogoFileName	Name of the image file containing the user's logo. File name can be up to 47 characters. The logo file name can be used to replace AudioCodes' default Web logo with a user defined logo. Use a gif, jpeg or jpg image file.
BkgImageFileName	Name of the file containing the user's background image. File name can be up to 47 characters. The background file can be used to replace AudioCodes' default background image with a user defined background. Use a gif, jpeg or jpg image file.

6.5.4 Security

Table 6-4: Security Parameters (continues on pages 227 to 228)

ini File Field Name Web Parameter Name	Valid Range and Description
EnableIPSec	For a description of this parameter, refer to Table 5-36 on page 176.
EnableMediaSecurity	For a description of this parameter, refer to Table 5-36 on page 176.
MediaSecurityBehaviour	For a description of this parameter, refer to Table 5-36 on page 176.
EnableSIPS	For a description of this parameter, refer to Table 5-3 on page 64.
TLSLocalSIPPort	For a description of this parameter, refer to Table 5-3 on page 64.
SIPSRequireClientCertificate	0 = The gateway doesn’t require client certificate (default). 1 = The gateway (when acting as a server for the TLS connection) requires reception of client certificate to establish the TLS connection. **Note:** The SIPS certificate files can be changed using the parameters ‘HTTPSCertFileName’ and ‘HTTPSRootFileName’.
Secure Hypertext Transport Protocol (HTTPS) Parameters	
HTTPSOnly	For a description of this parameter, refer to Table 5-36 on page 176.
HTTPSPort	Determine the local Secured HTTPS port of the device. The valid range is 1 to 65535 (other restrictions may apply within this range). The default port is 443.
WebAuthMode	For a description of this parameter, refer to Table 5-36 on page 176.
HTTPSRequireClientCertificate	Requires client certificates for HTTPS connection. The client certificate must be preloaded to the gateway, and its matching private key must be installed on the managing PC. Time and date must be correctly set on the gateway, for the client certificate to be verified. 0 = Client certificates are not required (default). 1 = Client certificates are required.
HTTPSRootFileName	Defines the name of the HTTPS trusted root certificate file to be loaded via TFTP. The file must be in base64-encoded PEM (Privacy Enhanced Mail) format. The valid range is a 47-character string. **Note:** This parameter is only relevant when the gateway is loaded via BootP/TFTP. For information on loading this file via the Embedded Web Server, refer to the Security section in the User’s Manual.
HTTPSPkeyFileName	Defines the name of a private key file (in unencrypted PEM format) to be loaded from the TFTP server.
HTTPSCertFileName	Defines the name of the HTTPS server certificate file to be loaded via TFTP. The file must be in base64-encoded PEM format. The valid range is a 47-character string. **Note:** This parameter is only relevant when the gateway is loaded via BootP/TFTP. For information on loading this file via the Embedded Web Server, refer to the Security section in the User’s Manual.
VoiceMenuPassword	For a description of this parameter, refer to Table 5-36 on page 176.
Internal Firewall Parameters	
AccessList_Source_IP	For a description of this parameter, refer to Table 5-35 on page 174.
AccessList_Net_Mask	For a description of this parameter, refer to Table 5-35 on page 174.
AccessList_Start_Port **AccessList_End_Port**	For a description of this parameter, refer to Table 5-35 on page 174.
AccessList_Protocol	For a description of this parameter, refer to Table 5-35 on page 174.
AccessList_Packet_Size	For a description of this parameter, refer to Table 5-35 on page 174.
AccessList_Byte_Rate	For a description of this parameter, refer to Table 5-35 on page 174.
AccessList_Byte_Burst	For a description of this parameter, refer to Table 5-35 on page 174.

Table 6-4: Security Parameters (continues on pages 227 to 228)

ini File Field Name Web Parameter Name	Valid Range and Description
AccessList_Allow_Type	For a description of this parameter, refer to Table 5-35 on page 174.
AccessList_MatchCount	For a description of this parameter, refer to Table 5-35 on page 174.

6.5.5 RADIUS Parameters

For detailed information on the supported RADIUS attributes, refer to Section 9.11 on page 288.

Table 6-5: RADIUS Parameter

ini File Field Name Web Parameter Name	Valid Range and Description
EnableRADIUS	For a description of this parameter, refer to Table 5-36 on page 176.
AAAIndications	0 = No indications (default). 3 = Accounting only
MaxRADIUSSessions	Number of concurrent calls that can communicate with the RADIUS server (optional). The valid range is 0 to 240. The default value is 240.
SharedSecret	For a description of this parameter, refer to Table 5-36 on page 176.
RADIUSRetransmission	Number of retransmission retries. The valid range is 1 to 10. The default value is 3.
RadiusTO	Determines the time interval (measured in seconds) the gateway waits for a response before a RADIUS retransmission is issued. The valid range is 1 to 30. The default value is 10.
RADIUSAuthServerIP	For a description of this parameter, refer to Table 5-36 on page 176.
RADIUSAuthPort	For a description of this parameter, refer to Table 5-36 on page 176.
RADIUSAccServerIP	IP address of accounting server.
RADIUSAccPort	Port number of Radius accounting server. The default value is 1646.
RadiusAccountingType	Determines when a RADIUS accounting report is issued. 0 = At the Release of the call only (default). 1 = At the Connect and Release of the call. 2 = At the Setup and Release of the call.
DefaultAccessLevel	For a description of this parameter, refer to Table 5-36 on page 176.
BehaviorUponRadiusTimeout	For a description of this parameter, refer to Table 5-36 on page 176.
RadiusLocalCacheMode	For a description of this parameter, refer to Table 5-36 on page 176.
RadiusLocalCacheTimeout	For a description of this parameter, refer to Table 5-36 on page 176.
RadiusVSAVendorID	For a description of this parameter, refer to Table 5-36 on page 176.
RadiusVSAAccessAttribute	For a description of this parameter, refer to Table 5-36 on page 176.

6.5.6 SNMP Parameters

Table 6-6: SNMP Parameters

ini File Field Name Web Parameter Name	Valid Range and Description
DisableSNMP	For a description of this parameter, refer to Table 5-37 on page 179.
SNMPPort	The device's local UDP port used for SNMP Get/Set commands. The range is 100 to 3999. The default port is 161.
SNMPTrustedMGR_x	Up to five IP addresses of remote trusted SNMP managers from which the SNMP agent accepts and processes get and set requests. **Note 1:** If no values are assigned to these parameters any manager can access the device. **Note 2:** Trusted managers can work with *all* community strings.
AlarmHistoryTableMaxSize	Determines the maximum number of rows in the Alarm History table. The parameter can be controlled by the Config Global Entry Limit MIB (located in the Notification Log MIB). The valid range is 50 to 1000. The default value is 500.
SNMP Trap Parameters	
SNMPManagerTableIP_x	For a description of this parameter, refer to Table 5-38 on page 181.
SNMPManagerTrapPort_x	For a description of this parameter, refer to Table 5-38 on page 181.
SNMPManagerTrapUser_x	This parameter can be set to the name of any configured SNMPV3 user to associate with this trap destination. This determines the trap format, authentication level, and encryption level. By default, the trap is associated with the SNMP trap community string.
SNMPManagerIsUsed_x	For a description of this parameter, refer to Table 5-38 on page 181.
SNMPManagerTrapSendingEnable_x	For a description of this parameter, refer to Table 5-38 on page 181.
SNMPTrapManagerHostName	For a description of this parameter, refer to Table 5-37 on page 179.
SNMP Community String Parameters	
SNMPReadOnlyCommunityString_x	For a description of this parameter, refer to Table 5-39 on page 182.
SNMPReadWriteCommunityString_x	For a description of this parameter, refer to Table 5-39 on page 182.
SNMPTrapCommunityString	For a description of this parameter, refer to Table 5-39 on page 182.
SNMP v3 Users Parameters	
SNMPUsers_Index	For a description of this parameter, refer to Table 5-40 on page 183.
SNMPUsers_Username	For a description of this parameter, refer to Table 5-40 on page 183.
SNMPUsers_AuthProtocol	For a description of this parameter, refer to Table 5-40 on page 183.
SNMPUsers_PrivProtocol	For a description of this parameter, refer to Table 5-40 on page 183.
SNMPUsers_AuthKey	For a description of this parameter, refer to Table 5-40 on page 183.
SNMPUsers_PrivKey	For a description of this parameter, refer to Table 5-40 on page 183.
SNMPUsers_Group	For a description of this parameter, refer to Table 5-40 on page 183.

6.5.7 SIP Configuration Parameters

Table 6-7: SIP Configuration Parameters (continues on pages 230 to 236)

ini File Field Name Web Parameter Name	Valid Range and Description
SIPTransportType	For a description of this parameter, refer to Table 5-3 on page 64.
TCPLocalSIPPort	For a description of this parameter, refer to Table 5-3 on page 64.
SIPDestinationPort	For a description of this parameter, refer to Table 5-3 on page 64.
EnableTCPConnectionReuse	For a description of this parameter, refer to Table 5-3 on page 64.
LocalSIPPort	For a description of this parameter, refer to Table 5-3 on page 64.
SIPGatewayName	For a description of this parameter, refer to Table 5-4 on page 72.
IsProxyUsed	For a description of this parameter, refer to Table 5-4 on page 72.
ProxyIP	For a description of this parameter, refer to Table 5-4 on page 72.
ProxyIP **ProxyIP** **ProxyIP**	For a description of this parameter, refer to Table 5-4 on page 72.
ProxyName	For a description of this parameter, refer to Table 5-4 on page 72.
EnableProxySRVQuery	This parameter is obsolete; use the parameter ProxyDNSQueryType.
EnableSRVQuery	This parameter is obsolete; use the parameter DNSQueryType.
AlwaysSendToProxy	For a description of this parameter, refer to Table 5-4 on page 72.
SendInviteToProxy	For a description of this parameter, refer to Table 5-4 on page 72.
PreferRouteTable	For a description of this parameter, refer to Table 5-4 on page 72.
EnableProxyKeepAlive	For a description of this parameter, refer to Table 5-4 on page 72.
ProxyKeepAliveTime	For a description of this parameter, refer to Table 5-4 on page 72.
DNSQueryType	For a description of this parameter, refer to Table 5-4 on page 72.
ProxyDNSQueryType	For a description of this parameter, refer to Table 5-4 on page 72.
UseSIPTgrp	For a description of this parameter, refer to Table 5-3 on page 64.
EnableGRUU	For a description of this parameter, refer to Table 5-3 on page 64.
UserAgentDisplayInfo	For a description of this parameter, refer to Table 5-3 on page 64.
SessionExpiresMethod	For a description of this parameter, refer to Table 5-3 on page 64.
UseGatewayNameForOptions	For a description of this parameter, refer to Table 5-4 on page 72.
IsProxyHotSwap	For a description of this parameter, refer to Table 5-4 on page 72.
ProxyHotSwapRtx	For a description of this parameter, refer to Table 5-4 on page 72.
ProxyRedundancyMode	For a description of this parameter, refer to Table 5-4 on page 72.
ProxyLoadBalancingMethod	For a description of this parameter, refer to Table 5-4 on page 72.
ProxyIPListRefreshTime	For a description of this parameter, refer to Table 5-4 on page 72.
IsFallbackUsed	For a description of this parameter, refer to Table 5-4 on page 72.
UserName	For a description of this parameter, refer to Table 5-4 on page 72.
Password	For a description of this parameter, refer to Table 5-4 on page 72.
Cnonce	For a description of this parameter, refer to Table 5-4 on page 72.
IsRegisterNeeded	For a description of this parameter, refer to Table 5-4 on page 72.
RegistrarIP	For a description of this parameter, refer to Table 5-4 on page 72.
RegistrarName	For a description of this parameter, refer to Table 5-4 on page 72.

Table 6-7: SIP Configuration Parameters (continues on pages 230 to 236)

ini File Field Name Web Parameter Name	Valid Range and Description
GWRegistrationName	For a description of this parameter, refer to Table 5-4 on page 72.
AuthenticationMode	For a description of this parameter, refer to Table 5-4 on page 72.
RegistrationTime	For a description of this parameter, refer to Table 5-4 on page 72.
RegistrationTimeDivider	For a description of this parameter, refer to Table 5-4 on page 72.
RegistrationRetryTime	For a description of this parameter, refer to Table 5-4 on page 72.
NumberOfActiveDialogs	Defines the maximum number of active SIP dialogs that are not call related (i.e., REGISTER and SUBSCRIBE). This parameter is used to control the Registration / Subscription rate. The valid range is 1 to 20. The default value is 20.
PrackMode	For a description of this parameter, refer to Table 5-3 on page 64.
AssertedIdMode	For a description of this parameter, refer to Table 5-3 on page 64.
UseTelURIForAssertedID	For a description of this parameter, refer to Table 5-3 on page 64.
EnableRPIheader	For a description of this parameter, refer to Table 5-3 on page 64.
IsUserPhone	For a description of this parameter, refer to Table 5-3 on page 64.
IsUserPhoneInFrom	For a description of this parameter, refer to Table 5-3 on page 64.
IsUseToHeaderAsCalledNumber	0 = Sets the destination number to the user part of the Request-URI for IP→Tel calls, and sets the 'Contact' header to the source number for Tel→ IP calls (default). 1 = Sets the destination number to the user part of the 'To' header for IP→Tel calls, and sets the 'Contact' header to the *username* parameter for Tel→IP calls.
EnableHistoryInfo	For a description of this parameter, refer to Table 5-3 on page 64.
SIPSubject	For a description of this parameter, refer to Table 5-3 on page 64.
MultiPtimeFormat	For a description of this parameter, refer to Table 5-3 on page 64.
EnableReasonHeader	For a description of this parameter, refer to Table 5-3 on page 64.
EnableSemiAttendedTransfer	For a description of this parameter, refer to Table 5-3 on page 64.
EnablePtime	0 = Remove the ptime header from SDP. 1 = Include the ptime header in SDP (default).
EnableUserInfoUsage	For a description of this parameter, refer to Table 5-7 on page 85.
EnableRport	Enables / disables the usage of the 'rport' parameter in the Via header. 0 = Enabled. 1 = Disabled (default). The gateway adds an 'rport' parameter to the Via header field of each outgoing SIP message. The first Proxy that receives this message sets the 'rport' value of the response to the actual port from which the request was received. This method is used, for example, to enable the gateway to identify its port mapping outside a NAT. If the Via doesn't include 'rport' tag, the destination port of the response will be taken from the host part of the VIA. If the Via includes 'rport' tag with no port value, the destination port of the response will be the source port of the incoming request. If the Via includes 'rport' tag with a port value (rport=1001), the destination port of the response will be the port indicated in the 'rport' tag.

Table 6-7: SIP Configuration Parameters (continues on pages 230 to 236)

ini File Field Name Web Parameter Name	Valid Range and Description
CoderName	Defines the gateway's coder list (Coders table). Up to five coders can be configured. Enter coders in the following format: CoderName=<Coder Name>,<Ptime>,<Rate>,<Payload Type>,<Silence Suppression Mode>. **Note 1:** The coder name is case-sensitive. **Note 2:** If silence suppression is not defined (for a specific coder), the value defined by the parameter EnableSilenceCompression is used. **Note 3:** The value of several fields is hard-coded according to well-known standards (e.g., the payload type of G.711 U-law is always 0). Other values can be set dynamically. If no value is specified for a dynamic field, a default value is assigned. If a value is specified for a hard-coded field, the value is ignored. **Note 4:** Only the ptime of the first coder in the defined coder list is declared in INVITE / 200 OK SDP, even if multiple coders are defined. **Note 5:** If the coder G.729 is selected and silence suppression is disabled (for this coder), the gateway includes the string 'annexb=no' in the SDP of the relevant SIP messages. If silence suppression is enabled or set to 'Enable w/o Adaptations', 'annexb=yes' is included. An exception to this logic is when the remote gateway is a Cisco device (IsCiscoSCEMode). **Note 6:** For a list of supported coders, refer to Table 5-5 on page 79. For example: CoderName = g711Alaw64k,20,,,0 CoderName = g711Ulaw64k,40 CoderName = g7231,90,1,,1 CoderName = g726,$$,$$,,0.
VBRCoderHeaderFormat	Defines the format of the RTP header for VBR coders. 0 = Payload only (no header, no TOC, no m-factor) (default). 1 = Supports RFC 2658 - 1 byte for interleaving header (always 0), TOC, no m-factor. 2 = Payload including TOC only, allow m-factor. 3 = RFC 3558.
TransparentCoderOnDataCall	0 = Only use coders from the coder list (default). 1 = Use transparent coder for data calls (according to RFC 4040). The 'Transparent' coder can be used on data calls. When the gateway receives a Setup message from the ISDN with 'TransferCapabilities = data', it can initiate a call using the coder 'Transparent' (even if the coder is not included in the coder list). The initiated INVITE includes the following SDP attribute: a=rtpmap:97 CLEARMODE/8000 The default Payload Type is set according to the CoderName table. If the Transparent coder is not set in the Coders table, the default value is set to 56. The Payload Type is negotiated with the remote side, i.e., the selected Payload Type is according to the remote side selection. The receiving gateway must include the 'Transparent' coder in its coder list.
IsFaxUsed	For a description of this parameter, refer to Table 5-3 on page 64.
T38UseRTPPort	Defines that the T.38 packets are sent / received using the same port as RTP packets. 0 = Use the RTP port +2 to send / receive T.38 packets (default). 1 = Use the same port as the RTP port to send / receive T.38 packets.
CngDetectorMode	For a description of this parameter, refer to Table 5-29 on page 152.
DefaultReleaseCause	For a description of this parameter, refer to Table 5-7 on page 85.
IPAlertTimeout	For a description of this parameter, refer to Table 5-3 on page 64.

Table 6-7: SIP Configuration Parameters (continues on pages 230 to 236)

ini File Field Name Web Parameter Name	Valid Range and Description
IPPSessionExpires	For a description of this parameter, refer to Table 5-3 on page 64.
MINSE	For a description of this parameter, refer to Table 5-3 on page 64.
SIPMaxRtx	For a description of this parameter, refer to Table 5-3 on page 64.
SipT1Rtx	For a description of this parameter, refer to Table 5-3 on page 64.
SipT2Rtx	For a description of this parameter, refer to Table 5-3 on page 64.
EnableEarlyMedia	For a description of this parameter, refer to Table 5-3 on page 64.
EnableTransfer	For a description of this parameter, refer to Table 5-8 on page 92.
XferPrefix	For a description of this parameter, refer to Table 5-8 on page 92.
EnableHold	For a description of this parameter, refer to Table 5-8 on page 92.
EnableForward	For a description of this parameter, refer to Table 5-8 on page 92.
EnableCallWaiting	For a description of this parameter, refer to Table 5-8 on page 92.
3WayConferenceMode	Defines the mode of operation when the 3-Way Conference feature is used. Valid options include: 0 = Conference-initiating INVITE (sent by the gateway), uses the ConferenceID concatenated with a unique identifier as the Request-UR (default). 1 = Conference-initiating INVITE (sent by the gateway), uses only the ConferenceID as the Reques-URI. If 3wayConferenceMode is set to 0, the Conference-initiating INVITE sent by the gateway, uses the ConferenceID concatenated with a unique identifier as the Request-URI. This same Request-URI is set as the Refer-To header value in the REFER messages that are sent to the two remote parties. If 3wayConferenceMode is set to 1, the Conference-initiating INVITE sent by the gateway, only uses the ConferenceID as the Reques-URI. The media server sets the Contact header of the 200 OK response to the actual unique identifier (Conference URI) to be used by the participants. This Conference URI is included (by the gateway), in the Refer-To header value in the REFER messages sent by the gateway to the remote parties. The remote parties join the conference by sending INVITE messages to the media server using this conference URI.
Enable3WayConference	For a description of this parameter, refer to Table 5-8 on page 92.
ConferenceCode	For a description of this parameter, refer to Table 5-8 on page 92.
ConferenceID	For a description of this parameter, refer to Table 5-8 on page 92.
BipOnConference	Determines whether a beep is played when a new participant joins a conference and when a participant leaves a conference (in the latter case, a beep of a different pitch is heard). 0 = Beep is disabled. 1 = Beep is enabled (default).
Send180ForCallWaiting	0 = Use 182 Queued response to indicate a call waiting (default). 1 = Use 180 Ringing response to indicate a call waiting.
HookFlashCode	For a description of this parameter, refer to Table 5-8 on page 92.
UseSIPURIForDiversionHeader	Sets the URI format in the Diversion header. 0 = 'tel:' (default). 1 = 'sip:'.
WarningToneDuration	Defines the duration (in seconds) for which Off-Hook Warning Tone is played to the user. The valid range is -1 to 2,147,483,647 seconds. The default is 600 seconds. **Note**: A negative value indicates that the tone is played infinitely.

Table 6-7: SIP Configuration Parameters (continues on pages 230 to 236)

***ini* File Field Name** **Web Parameter Name**	**Valid Range and Description**
FirstCallWaitingToneID	Determines the index of the first Call Waiting Tone in the CPT file. This feature enables the called party to distinguish between four different call origins (e.g., external vs. internal calls). The gateway plays the tone received in the 'play tone CallWaitingTone#' parameter of an INFO message + the value of this parameter - 1. The valid range is -1 to 100. The default value is -1 (not used). **Note 1:** It is assumed that all Call Waiting Tones are defined in sequence in the CPT file. **Note 2:** This feature is relevant only to Broadsoft's application servers (the tone is played using INFO message).
RTPOnlyMode	Enables the gateway to start sending and/or receiving RTP packets to and from remote endpoints without the need to establish a Control session. The remote IP address is determined according to the Tel2IP Routing table. The port is the same port as the local RTP port (set by BaseUDPPort and the channel on which the call was received). Valid options include: ▪ 0 = Disable (default). ▪ 1 = Transmit & Receive. ▪ 2 = Transmit Only. ▪ 3 = Receive Only.
RxDTMFOption	For a description of this parameter, refer to Table 5-6 on page 80.
TxDTMFOption	For a description of this parameter, refer to Table 5-6 on page 80.
DisableAutoDTMFMute	Enables / disables the automatic mute of DTMF digits when out-of-band DTMF transmission is used. 0 = Auto mute is used (default). 1 = No automatic mute of in-band DTMF. When 'DisableAutoDTMFMute=1', the DTMF transport type is set according to the parameter 'DTMFTransportType' and the DTMF digits aren't muted if out-of-band DTMF mode is selected ('TxDTMFOption =1, 2 or 3'). This enables the sending of DTMF digits in-band (transparent of RFC 2833) in addition to out-of-band DTMF messages. **Note:** Usually this mode is not recommended.
EnableReasonHeader	For a description of this parameter, refer to Table 5-3 on page 64.
3xxBehavior	For a description of this parameter, refer to Table 5-3 on page 64.
MaxActiveCalls	For a description of this parameter, refer to Table 5-7 on page 85.
MaxCallDuration	For a description of this parameter, refer to Table 5-7 on page 85.
EnableBusyOut	For a description of this parameter, refer to Table 5-7 on page 85.
EnableDigitDelivery2IP	For a description of this parameter, refer to Table 5-7 on page 85.
EnableDigitDelivery	For a description of this parameter, refer to Table 5-7 on page 85.
[Module n] **Authentication_x** n = 0 or 5 (module serial number)	Authentication_<Port> = <Username>,<Password> Port = 0 to 3. For example: [Module 1] Authentication_0 = david,14325 Authentication_1 = Alex,18552 **Note 1:** Using the sign '$$' enables the user to omit either the username or the password. For instance, Authentication_5 = $$, 152. In this case, endpoint 5's phone number is used instead of username. **Note 2:** To configure the authentication username and password using the Web interface, refer to Section 5.5.6.1 on page 120.

Table 6-7: SIP Configuration Parameters (continues on pages 230 to 236)

ini File Field Name Web Parameter Name	Valid Range and Description
SITDetectorEnable	Enables or disables Special Information Tone (SIT) detection according to the ITU-T recommendation E.180/Q.35. Valid options include: 0 = Disable (default). 1 = Enable.
AMDTimeout	Timeout (in msec) between receiving CONNECT messages from the ISDN and sending of Answering Machine Detection (AMD) results. The valid range is 1 to 30,000. The default is 2,000 (2 seconds).
Profile Parameters	
CoderName_ID	Defines groups of coders that can be associated with IP or Tel profiles (up to five coders in each group). Enter coder groups in the following format: CoderName_<coder group ID from 1 to 4>=<Coder Name>,<Ptime>,<Rate>,<Payload Type>,<Silence Suppression Mode>. **Note 1:** The coder name is case-sensitive. **Note 2:** For a list of coders supported by the gateway, refer to Table 5-5 on page 79. **Note 3:** If silence suppression is not defined (for a specific coder), the value defined by the parameter EnableSilenceCompression is used. **Note 4:** The value of several fields is hard-coded according to well-known standards (e.g., the payload type of G.711 U-law is always 0). Other values can be set dynamically. If no value is specified for a dynamic field, a default value is assigned. If a value is specified for a hard-coded field, the value is ignored. **Note 5:** Only the ptime of the first coder in the defined coder list is declared in INVITE / 200 OK SDP, even if multiple coders are defined. **Note 6:** If the coder G.729 is selected and silence suppression is enabled (for this coder), the gateway includes the string ‘annexb=no’ in the SDP of the relevant SIP messages. If silence suppression is set to ‘Enable w/o Adaptations’, ‘annexb=yes’ is included. An exception to this logic is when the remote gateway is a Cisco device (IsCiscoSCEMode). **Note 7:** This parameter (CoderName_ID) can appear up to 20 times (five coders in four coder groups). For example: CoderName_1 = g711Alaw64k,20,,,0 CoderName_1 = g711Ulaw64k,40 CoderName_1 = g7231,90,1,,1 CoderName_2 = g726,$$,2,,0

Table 6-7: SIP Configuration Parameters (continues on pages 230 to 236)

ini File Field Name Web Parameter Name	Valid Range and Description
IPProfile_ID [IP Profile Settings]	IPProfile_<Profile ID> = <Profile Name>,<Preference>,<Coder Group ID>,<IsFaxUsed*>, <DJBufMinDelay *>, <DJBufOptFactor *>,<IPDiffServ*>, <ControlIPDiffServ*>,<N/A use $$ instead>, <RTPRedundancyDepth>,<RemoteBaseUDPPort>,<CNGmode>, <VxxTransportType>, <NSEMode>,<N/A use $$ instead>, <PlayRBTone2IP>, <EnableEarlyMedia*>, <ProgressIndicator2IP*> Preference = (1-20) The preference option is used to determine the priority of the Profile. Where '20' is the highest preference value. If both IP and Tel profiles apply to the same call, the coders and other common parameters (noted by an asterisk) of the preferred Profile are applied to that call. If the Preference of the Tel and IP Profiles is identical, the Tel Profile parameters are applied. For example: IPProfile_1 = name1,2,1,0,10,13,15,44,1,1,6000,0,2,0,0,0,1,0 IPProfile_2 = name2,$$,$$,$$,$$,$$,$$,$$,$$,1,$$,$$,$$,$$,$$,$$,$$,$$ $$ = Not configured, the default value of the parameter is used. (*) = Common parameter used in both IP and Tel profiles. **Note 1:** The IP ProfileID can be used in the Tel2IP and IP2Tel routing tables (Prefix and PSTNPrefix parameters). **Note 2:** 'Profile Name' assigned to a ProfileID, enabling user to identify it intuitively and easily. **Note 3:** This parameter can appear up to 9 times (ID = 1 to 9).
TelProfile_ID [Tel Profile Settings]	TelProfile_<Profile ID> = <Profile Name>,<Preference>,<Coder Group ID>,<IsFaxUsed *>,<DJBufMinDelay *>, <DJBufOptFactor *>,<IPDiffServ *>,<ControlIPDiffServ*>,<DtmfVolume>,<InputGain>, <VoiceVolume>,<EnableReversePolarity>,<EnableCurrentDisconnect>, <EnableDigitDelivery>, <ECE>, <MWIAnalogLamp>, <MWIDisplay>, <FlashHookPeriod>, <EnableEarlyMedia*>, <ProgressIndicator2IP*> Preference = (1-10) The preference option is used to determine the priority of the Profile. Where '20' is the highest preference value. If both IP and Tel profiles apply to the same call, the coders and other common parameters (noted by an asterisk) of the preferred Profile are applied to that call. If the Preference of the Tel and IP Profiles is identical, the Tel Profile parameters are applied. For examples: TelProfile_1 = FaxProfile,1,1,1,40,13,22,33,$$,$$,$$,0,0,0,1,0,0,$$,0,$$ TelProfile_2 = ModemProfile,2,2,0,40,13,$$,$$,$$,$$,$$,$$,$$,$$,0,0,0,$$,0,$$ $$ = Not configured, the default value of the parameter is used. (*) = Common parameter used in both IP and Tel profiles. **Note 1:** The Tel ProfileID can be used in the Trunk Group table (TrunkGroup_x parameter). **Note 2:** 'Profile Name' assigned to a ProfileID, enabling users to identify it intuitively and easily. **Note 3:** This parameter can appear up to 9 times (ID = 1 to 9

6.5.8 Voice Mail Parameters

For detailed information on the Voice Mail (VM) application, refer to the CPE Configuration Guide for Voice Mail.

Table 6-8: Voice Mail Configuration Parameters

ini File Field Name Web Parameter Name	Valid Range and Description
VoiceMailInterface	For a description of this parameter, refer to Table 5-19 on page 130.
SubscriptionMode	For a description of this parameter, refer to Table 5-4 on page 72.
LineTransferMode	For a description of this parameter, refer to Table 5-19 on page 130.
WaitForDialTime	For a description of this parameter, refer to Table 5-18 on page 127.
MWIOnCode	For a description of this parameter, refer to Table 5-19 on page 130.
MWIOffCode	For a description of this parameter, refer to Table 5-19 on page 130.
TelDisconnectCode	For a description of this parameter, refer to Table 5-19 on page 130.
The following digit pattern parameters apply only to VM applications that use the DTMF communication method. For the available pattern syntaxes, refer to the CPE Configuration Guide for Voice Mail.	
DigitPatternForwardOnBusy	For a description of this parameter, refer to Table 5-19 on page 130.
DigitPatternForwardOnNoAnswer	For a description of this parameter, refer to Table 5-19 on page 130.
DigitPatternForwardOnDND	For a description of this parameter, refer to Table 5-19 on page 130.
DigitPatternForwardNoReason	For a description of this parameter, refer to Table 5-19 on page 130.
DigitPatternInternalCall	For a description of this parameter, refer to Table 5-19 on page 130.
DigitPatternExternalCall	For a description of this parameter, refer to Table 5-19 on page 130.

6.5.9 ISDN and CAS Interworking-Related Parameters

Table 6-9: ISDN and CAS Interworking-Related Parameters (continues on pages 237 to 242)

ini File Field Name Web Parameter Name	Valid Range and Description
EnableTDMoverIP	For a description of this parameter, refer to Table 5-22 on page 136.
EnableISDNTunnelingTel2IP	For a description of this parameter, refer to Table 5-22 on page 136.
EnableISDNTunnelingIP2Tel	For a description of this parameter, refer to Table 5-22 on page 136.
ISDNDuplicateQ931BuffMode	Controls the activation / deactivation of delivering raw Q.931 messages. 0 = ISDN messages aren't duplicated (default). 128 = All ISDN messages are duplicated. **Note:** This parameter is not updated on-the-fly and requires a gateway reset.
EnableQSIGTunneling	For a description of this parameter, refer to Table 5-22 on page 136.

Table 6-9: ISDN and CAS Interworking-Related Parameters (continues on pages 237 to 242)

ini File Field Name Web Parameter Name	Valid Range and Description
PlayRBTone2Trunk_ID	ID = Trunk number (0-7). 0 (Don't play) = The ISDN / CAS gateway doesn't play a Ringback Tone (RBT). No PI is sent to the ISDN, unless the parameter 'Progress Indicator to ISDN' is configured differently. 1 (Play) = The CAS gateway plays a local RBT to PSTN after receipt of a 180 ringing response (with or without SDP). **Note:** Reception of a 183 response doesn't cause the CAS gateway to play an RBT (unless 'SIP183Behavior = 1'). The ISDN gateway functions according to the parameter 'LocalISDNRBSource': ▪ If the ISDN gateway receives a 180 ringing response (with or without SDP) and 'LocalISDNRBSource = 1', it plays a RBT and sends an Alert with PI = 8 (unless the parameter 'Progress Indicator to ISDN' is configured differently). ▪ If 'LocalISDNRBSource = 0', the ISDN gateway doesn't play an RBT and an Alert message (without PI) is sent to the ISDN. In this case, the PBX / PSTN should play the RBT to the originating terminal by itself. **Note:** Reception of a 183 response doesn't cause the ISDN gateway to play an RBT; the gateway issues a Progress message (unless 'SIP183Behavior = 1'). If 'SIP183Behavior = 1', the 183 response is treated the same way as a 180 ringing response. 2 = Play according to 'early media' (default). If a 180 response is received and the voice channel is already open (due to a previous 183 early media response or due to an SDP in the current 180 response), the ISDN / CAS gateway doesn't play the RBT; PI = 8 is sent in an ISDN Alert message (unless the parameter 'Progress Indicator to ISDN' is configured differently). If a 180 response is received but the 'early media' voice channel is not opened, the CAS gateway plays an RBT to the PSTN; the ISDN gateway functions according to the parameter 'LocalISDNRBSource': ▪ If 'LocalISDNRBSource = 1', the ISDN gateway plays an RBT and sends an ISDN Alert with PI = 8 to the ISDN (unless the parameter 'Progress Indicator to ISDN' is configured differently). ▪ If 'LocalISDNRBSource = 0', the ISDN gateway doesn't play an RBT. No PI is sent in the ISDN Alert message (unless the parameter 'Progress Indicator to ISDN' is configured differently). In this case, the PBX / PSTN should play an RBT tone to the originating terminal by itself. **Note:** Reception of a 183 response results in an ISDN Progress message (unless 'SIP183Behavior = 1'). If 'SIP183Behavior = 1' (183 is handled in the same way as a 180+SDP), the gateway sends an Alert message with PI = 8, without playing an RBT.
PlayRBTone2Tel	For a description of this parameter, refer to Table 5-3 on page 64.
PlayRBTone2IP	For a description of this parameter, refer to Table 5-3 on page 64.
DefaultCauseMapISDN2IP	Defines a single default ISDN Release Cause that is used (in ISDN to IP calls) instead of all received release causes except when the following Q.931 cause values are received: Normal Call Clearing (16), User Busy (17), No User Responding (18) or No Answer from User (19). The range is valid Q.931 release causes (0 to 127). The default value is 0 (indicates that the parameter is not configured - static mapping is used).

Table 6-9: ISDN and CAS Interworking-Related Parameters (continues on pages 237 to 242)

ini File Field Name Web Parameter Name	Valid Range and Description
CauseMapSIP2ISDN_ID	Defines a flexible mapping of SIP Responses and Q.850 Release Causes. CauseMapSIP2ISDN_<ID> = <SIP Response>,<Q.850 Release Cause> When a SIP response is received (from the IP side), the gateway searches this mapping table for a match. If the SIP response is found, the Release Cause assigned to it is sent to the PSTN. If no match is found, the default static mapping is used. For example: CauseMapSIP2ISDN=404,3 **Note:** This parameter can appear up to 12 times.
CauseMapISDN2SIP_ID	Defines a flexible mapping of Q.850 Release Causes and SIP Responses. CauseMapISDN2SIP_<ID> = <Q.850 Release Cause>,<SIP Response> When a Release Cause is received (from the PSTN side), the gateway searches this mapping table for a match. If the Q.850 Release Cause is found, the SIP response assigned to it is sent to the IP side. If no match is found, the default static mapping is used. For example: CauseMapISDN2SIP=6,406 **Note:** This parameter can appear up to 12 times.
RemoveCLIWhenRestricted	For a description of this parameter, refer to Table 5-22 on page 136.
ScreeningInd2ISDN	For a description of this parameter, refer to Table 5-22 on page 136.
ProgressIndicator2ISDN_ID	ID = Trunk number (0-7). 0, 1 or 8 -1 = Not configured (default). If set to ‘0’ PI is not sent to ISDN If set to ‘1’ or ‘8’ the PI value is sent to PSTN in Q.931/Proceeding and Alerting messages. If not configured, the PI in ISDN messages is set according to the ‘Play Ringback to Tel’ parameter. Usually if PI = 1 or 8, the PSTN/PBX cuts through the audio channel without playing local Ringback tone, enabling the originating party to hear remote Call Progress Tones or network announcements
ProgressIndicator2IP	For a description of this parameter, refer to Table 5-7 on page 85.
PIForDisconnectMsg_ID	ID = Trunk number (0-7). Defines the gateway’s behavior when a Disconnect message is received from the ISDN before a Connect message was received. -1 (Not configured) = Sends a 183 message according to the received PI in the ISDN Disconnect message. If PI = 1 or 8, the gateway sends a 183 response, enabling the PSTN to play a voice announcement to the IP side. If there isn’t a PI in the Disconnect message, the call is released (default). 0 = Do not send a 183 message to IP. The call is released. 1, 8 = Sends 183 message to IP.
ConnectOnProgressInd	0 = Connect message isn’t sent after 183 Session Progress is received (default). 1 = Connect message is sent after 183 Session Progress is received. This feature enables the play of announcements from IP to PSTN without the need to answer the Tel→IP call. It can be used with PSTN networks that don’t support the opening of a TDM channel before an ISDN Connect message is received.
SIP183Behavior	For a description of this parameter, refer to Table 5-3 on page 64.

Table 6-9: ISDN and CAS Interworking-Related Parameters (continues on pages 237 to 242)

ini File Field Name Web Parameter Name	Valid Range and Description
LocalISDNRBSource_ID	ID = Trunk number (0-7). Determines whether Ringback tone is played to the ISDN by the PBX / PSTN or by the gateway. 0 = PBX / PSTN (default). 1 = Gateway. This parameter is applicable to ISDN protocols. It is used simultaneously with the parameter 'PlayRBTone2Trunk'.
PSTNAlertTimeout	For a description of this parameter, refer to Table 5-22 on page 136.
ISDNTransferCapability_ID	ID = Trunk number. Defines the IP→ISDN Transfer Capability of the Bearer Capability IE in ISDN Setup messages. 0 = Audio 3.1 (default). 1 = Speech. 2 = Data. **Note:** If this parameter isn't configured or equals to '–1', Audio 3.1 capability is used.
SendISDNTransferOnConnect	For a description of this parameter, refer to Table 5-22 on page 136.
ISDNSubAddressFormat	Determines the format of the Subaddress value for ISDN Calling and Called numbers. 0 = ASCII (default). 1 = BCD. For IP-to-Tel calls, if the incoming INVITE message includes 'Subaddress' values for the Called Number (in the Request-URI) and/or the Calling Number (in the From header), these values are interworked to the outgoing ISDN Setup message. If the incoming ISDN SETUP message includes 'subaddress' values for the Called Number and/or the Calling Number, these values are interworked to the outgoing SIP INVITE message.
EnableUUITel2IP	For a description of this parameter, refer to Table 5-22 on page 136.
EnableUUIIP2Tel	For a description of this parameter, refer to Table 5-22 on page 136.
ScreeningInd2IP	For a description of this parameter, refer to Table 5-22 on page 136.
SupportRedirectInFacility	0 = Not Supported (default). 1 = Supports partial retrieval of Redirect Number (number only) from a Facility IE in ISDN Setup messages. Applicable to Redirect number according to ECMA-173 Call Diversion Supplementary Services. **Note:** To enable this feature, 'ISDNDuplicateQ931BuffMode' must be set to 1.
EnableCIC	0 = Do not relay the Carrier Identification Code (CIC) to ISDN (default). 1 = CIC is relayed to ISDN in Transit Network Selection IE. If enabled, the CIC code (received in an INVITE Request-URI) is included in a TNS IE in ISDN Setup message. For example: INVITE sip:555666;cic=2345@100.2.3.4 sip/2.0. **Note:** Currently this feature is supported only in SIP→ISDN direction.
EnableAOC	0 = Not used (default). 1 = ISDN Advice of Charge (AOC) messages are interworked to SIP. The gateway supports reception of ISDN (Euro ISDN) AOC messages. AOC messages can be received during a call (Facility messages) or at the end of a call (Disconnect or Release messages). The gateway converts the AOC messages into SIP INFO (during a call) and BYE (end of a call) messages using a proprietary AOC SIP header. The gateway supports both Currency and Pulse AOC messages.
TimeForReorderTone	For a description of this parameter, refer to Table 5-18 on page 127.

Table 6-9: ISDN and CAS Interworking-Related Parameters (continues on pages 237 to 242)

ini File Field Name Web Parameter Name	Valid Range and Description
DisconnectOnBusyTone	For a description of this parameter, refer to Table 5-18 on page 127.
PlayBusyTone2ISDN	This parameter enables the Mediant 1000 ISDN gateway to play a Busy or a Reorder tone to the PSTN after a call is released. 0 = Immediately sends an ISDN Disconnect message (default). 1 = Sends an ISDN Disconnect message with PI=8 and plays a Busy or a Reorder tone to the PSTN (depending on the release cause). 2 = Delays the sending of an ISDN Disconnect message for 'TimeForReorderTone' seconds and plays a Busy or a Reorder tone to the PSTN. Applicable only if the call is released from the IP before it reaches the Connect state. Otherwise, the Disconnect message is sent immediately and no tones are played.
TrunkTransferMode_X	0 = Not supported (default). 1 = Supports CAS NFA DMS-100 transfer. When a SIP REFER message is received, the gateway performs a Blind Transfer by executing a CAS Wink, waits for an acknowledge Wink from the remote side, dials the Refer-to number to the switch and then releases the call. **Note:** A specific NFA CAS table is required. 2 = Supports ISDN transfer (RLT / TBCT / ECT). When a SIP REFER message is received, the gateway performs a transfer by sending FACILITY messages to the PBX with the necessary information on the call's legs that are to be connected. The different ISDN variants use slightly different methods (using FACILITY messages) to perform the transfer. 3 = Supports CAS Normal transfer. When a SIP REFER message is received, the gateway performs a Blind Transfer by executing a CAS Wink, dialing the Refer-to number to the switch and then releasing the call.
CASTransportType	For a description of this parameter, refer to Table 5-28 on page 149.
CASAddressingDelimiters	Determines if delimiters are added to the dialed address or dialed ANI parameters. Valid options include: 0 = Disable (default) 1 = Enable When this parameter is enabled, delimiters such as '*', '#', and 'ST' are added to the dialed address or dialed ANI parameters. When it is disabled, the address and ANI strings remain without delimiters.
CASDelimitersPaddingUsage	Defines the digits string delimiter padding usage per trunk. 0 (default) = default address string padding: '*XXX#' (where XXX is the digit string that begins with '*' and ends with '#', when using padding). 1 = special use of asterisks delimiters: '*XXX*YYY*' (where XXX is the address, YYY is the source phone number, and '*' is the only delimiter padding).
EnableVoiceDetection	For a description of this parameter, refer to Table 5-18 on page 127.
EnableDSPIPMDetectors	Enables / disables the gateway's DSP detectors. 0 = Disable (default). 1 = Enable. **Note1:** The gateway's Feature Key should contain the "IPMDetector" DSP option. **Note 2:** When EnableDSPIPMDetectors = 1, the number of available gateway channels is reduced.
XChannelHeader	For a description of this parameter, refer to Table 5-7 on page 85.
AddIEinSetup	For a description of this parameter, refer to Table 5-22 on page 136.
SendIEonTG	For a description of this parameter, refer to Table 5-22 on page 136.

Table 6-9: ISDN and CAS Interworking-Related Parameters (continues on pages 237 to 242)

ini File Field Name Web Parameter Name	Valid Range and Description
ISDNDMSTimerT310	Overrides the T310 timer for the DMS-100 ISDN variant. T310 defines the timeout between the reception of Proceeding message and the reception of Alert / Connect message. The valid range is 10 to 30. The default value is 10 (seconds). **Note:** Applicable only to Nortel DMS and Nortel MERIDIAN PRI variants (ProtocolType = 14 and 35).
ISDNJapanNTTTimerT3JA	T3_JA timer (in seconds). This parameter overrides the internal PSTN T301 timeout on the Users Side (TE side). If an outgoing call from the Mediant 1000 to ISDN is not answered during this timeout, the call is released. The valid range is 10 to 240. The default value is 50. Applicable only to Japan NTT PRI variant (ProtocolType = 16). **Note:** This timer is also affected by the parameter 'PSTNAlertTimeout'.
EnablePatternDetector	Enables or disables activation of the Pattern Detector (PD). Valid options include: 0 = Disable (default). 1 = Enable.
PDPattern	Defines the patterns that can be detected by the Pattern Detector. The valid range is 0 to 0xFF.
PDThreshold	Defines the number of consecutive patterns to trigger the pattern detection event. The valid range is 0 to 31. The default is 5.
EarlyAnswerTimeout	Defines the time (in seconds) the gateway waits for a CONNECT response from the called party (Tel side) after sending a SETUP message. If the timer expires, the call is answered by sending a 200 OK message (IP side). The valid range is 0 to 600. The default value is 0 (disable).

6.5.10 Analog Telephony Parameters

Table 6-10: Analog Telephony Parameters (continues on pages 242 to 246)

ini File Field Name Web Parameter Name	Valid Range and Description
FXONumberOfRings	Defines the number of rings before the FXO gateway answers a call. The valid range is 0 to 255. The default is 0 seconds.
ChargeCode	ChargeCode_<Charge Code ID> = <1st period end time>,<1st period pulse interval>,<1st period pulses on answer>, <2nd period end time>,<2nd period pulse interval>,<2nd period pulses on answer>, <3rd period end time>,<3rd period pulse interval>,<3rd period pulses on answer>, <4th period end time>,<4th period pulse interval>,<4th period pulses on answer> For example: ChargeCode_1 = 07,30,1,14,20,2,20,15,1,00,60,1 ChargeCode_2 = 05,60,1,14,20,1,00,60,1 ChargeCode_3 = 00,60,1 **Note 1:** Up to 25 different metering rules can be defined by repeating the parameter 25 times. **Note 2:** To configure the Charge Codes table using the Web interface, refer to Section 5.5.2.3.1 on page 96.

Table 6-10: Analog Telephony Parameters (continues on pages 242 to 246)

***ini* File Field Name** **Web Parameter Name**	**Valid Range and Description**
[Module n] **TargetOfChannelX** n = 0 or 5 (module serial number)	TargetOfChannel<Port> = <Phone>,<Mode> Port = 0 to 3. For example: [Module 0] TargetOfChannel0 = 1001,1 TargetOfChannel3 = 911,2 **Note 1:** The numbering of channels starts with 0. **Note 2:** Define this parameter for each gateway port you want to use for Automatic Dialing. **Note 3:** To configure the Automatic Dialing Table using the Web interface, refer to Section 5.5.6.2 on page 122.
[Module n] **CallerDisplayInfoX** n = 0 or 5 (module serial number)	CallerDisplayInfo<Port> = <Caller ID string>,<Restriction> Port = 0 to 3. 0 = Not restricted (default). 1 = Restricted. For example: [Module 0] CallerDisplayInfo0 = Susan C.,0 CallerDisplayInfo2 = Mark M.,1 **Note 1:** The numbering of channels starts with 0. **Note 2:** To configure the Caller Display Information table using the Web interface, refer to Section 5.5.6.3 on page 123.
[Module n] **FwdInfo_x** n = 0 or 5 (module serial number)	FwdInfo_<Port (0 to 23)> = <Forward Type>, <Forwarded SIP User Identification>, <Timeout (in seconds) for No Reply> For example: [Module 0] FwdInfo_0 = 1,1001 FwdInfo_1 = 1,2003@10.5.1.1 FwdInfo_2 = 3,2005,30 **Note 1:** The numbering of gateway ports starts with 0. **Note 2:** To configure the Call Forward table using the Web interface, refer to Section 5.5.6.4 on page 124.
[Module n] **EnableCallerID_X** n = 0 or 5 (module serial number)	EnableCallerID_<Port> = <Caller ID> Port = 0 to 3. Caller ID: 0 = Disabled (default). 1 = Enabled. If not configured, use the global parameter 'EnableCallerID'. **Note 1:** The numbering of ports starts with 0. **Note 2:** To configure the Call ID Permissions table using the Web interface, refer to Section 5.5.6.5 on page 125.
EnableDIDWink	For a description of this parameter, refer to Table 5-7 on page 85.
DelayBeforeDIDWink	For a description of this parameter, refer to Table 5-7 on page 85.
EnableReversalPolarity	For a description of this parameter, refer to Table 5-7 on page 85.
EnableCurrentDisconnect	For a description of this parameter, refer to Table 5-7 on page 85.
TelConnectCode	For a description of this parameter, refer to Table 5-7 on page 85.
CutThrough	For a description of this parameter, refer to Table 5-7 on page 85.
FXSOOSBehavior	For a description of this parameter, refer to Table 5-7 on page 85.

Table 6-10: Analog Telephony Parameters (continues on pages 242 to 246)

ini File Field Name Web Parameter Name	Valid Range and Description
NumberOfWaitingIndications	For a description of this parameter, refer to Table 5-8 on page 92.
TimeBetweenWaitingIndications	For a description of this parameter, refer to Table 5-8 on page 92.
TimeBeforeWaitingIndication	For a description of this parameter, refer to Table 5-8 on page 92.
WaitingBeepDuration	For a description of this parameter, refer to Table 5-8 on page 92.
EnableCallerID	For a description of this parameter, refer to Table 5-8 on page 92.
CallerIDType	For a description of this parameter, refer to Table 5-8 on page 92.
EnableMWI	For a description of this parameter, refer to Table 5-8 on page 92.
MWIAnalogLamp	For a description of this parameter, refer to Table 5-8 on page 92.
MWIDisplay	For a description of this parameter, refer to Table 5-8 on page 92.
EnableMWISubscription	For a description of this parameter, refer to Table 5-8 on page 92.
MWIServerIP	For a description of this parameter, refer to Table 5-8 on page 92.
SubscribeRetryTime	For a description of this parameter, refer to Table 5-8 on page 92.
MWIExpirationTime	For a description of this parameter, refer to Table 5-8 on page 92.
StutterToneDuration	For a description of this parameter, refer to Table 5-8 on page 92.
PayPhoneMeteringMode	For a description of this parameter, refer to Table 5-9 on page 95.
MeteringType	For a description of this parameter, refer to Table 5-9 on page 95.
KeyCFUnCond	For a description of this parameter, refer to Table 5-10 on page 98.
KeyCFNoAnswer	For a description of this parameter, refer to Table 5-10 on page 98.
KeyCFBusy	For a description of this parameter, refer to Table 5-10 on page 98.
KeyCFBusyOrNoAnswer	For a description of this parameter, refer to Table 5-10 on page 98.
KeyCFDoNotDisturb	For a description of this parameter, refer to Table 5-10 on page 98.
KeyCFDeact	For a description of this parameter, refer to Table 5-10 on page 98.
KeyCLIR	For a description of this parameter, refer to Table 5-10 on page 98.
KeyCLIRDeact	For a description of this parameter, refer to Table 5-10 on page 98.
KeyHotLine	For a description of this parameter, refer to Table 5-10 on page 98.
KeyHotLineDeact	For a description of this parameter, refer to Table 5-10 on page 98.
IsTwoStageDial	For a description of this parameter, refer to Table 5-18 on page 127.
IsWaitForDialTone	For a description of this parameter, refer to Table 5-18 on page 127.
FXOBetweenRingTime	For a description of this parameter, refer to Table 5-18 on page 127.
RingsBeforeCallerID	For a description of this parameter, refer to Table 5-18 on page 127.
DisconnectOnDialTone	For a description of this parameter, refer to Table 5-18 on page 127.
GuardTimeBetweenCalls	For a description of this parameter, refer to Table 5-18 on page 127.
NTTDIDSignallingForm	Determines the type of Direct Inward Dialing (DID) signaling support for NTT (Japan) modem: DTMF- or Frequency Shift Keying (FSK)-based signaling. Gateways can be connected to Japan's NTT PBX using 'Modem' DID lines. These DID lines are used to deliver a called number to the PBX. Valid options include: 0 = FSK-based signaling (default) 1 = DTMF-based signaling **Note:** Applicable only to FXS modules.

Table 6-10: Analog Telephony Parameters (continues on pages 242 to 246)

ini File Field Name Web Parameter Name	Valid Range and Description
EnableDID	Enables Japan NTT 'Modem' Direct Inward Dialing (DID) support. FXS modules can be connected to Japan's NTT PBX using 'Modem' DID lines. These DID lines are used to deliver a called number to the PBX (applicable to FXS modules). The DID signal can be sent alone or combined with an NTT Caller ID signal.
EnableDID_X	Enables generation of Japan NTT Modem DID signal per port. EnableDID_<Port> = <Modem DID> Modem DID: 0 = Disabled (default). 1 = Enabled. If not configured, use the global parameter 'EnableDID'. **Note:** Applicable only to FXS modules.
EnableCallerIDTypeTwo	Disables the generation of Caller ID type 2 when the phone is offhooked. 0 = Caller ID type 2 isn't played. 1 = Caller ID type 2 is played (default).
PolarityReversalType	Defines the voltage change slope during polarity reversal or wink. 0 = Soft (default). 1 = Hard. **Note 1:** Some Caller ID signals use reversal polarity and/or wink signals. In these cases it is recommended to set PolarityReversalType to 1 (Hard). **Note 2:** Applicable only to FXS modules.
CurrentDisconnectDuration	Duration of the current disconnect pulse (in msec). The default is 900 msec, The range is 200 to 1500 msec. Applicable for both FXS and FXO modules. **Note:** The FXO modules' detection range is +/-200 msec of the parameter's value + 100. For example if CurrentDisconnectDuration = 200, the detection range is 100 to 500 msec.
CurrentDisconnectDefaultThreshold	Determines the line voltage threshold which, when reached, is considered a current disconnect detection. The valid range is 0 to 20 Volts. The default value is 4 Volts. **Note:** Applicable only to FXO modules.
TimeToSampleAnalogLineVoltage	Determines the frequency at which the analog line voltage is sampled (after offhook), for detection of the current disconnect threshold. The valid range is 100 to 2500 msec. The default value is 1000 msec. **Note:** Applicable only to FXO modules.
AnalogCallerIDTimimgMode	0 = Caller ID is generated between the first two rings (default). 1 = The gateway attempts to find an optimized timing to generate the Caller ID according to the selected Caller ID type. Note that when used with distinctive ringing, the Caller ID signal doesn't change the distinctive ringing timing. **Note:** Applicable only to FXS modules.
BellcoreCallerIDTypeOneSubStandard	Selects the Bellcore Caller ID sub-standard. 0 = Between rings (default). 1 = Not ring related.
ETSICallerIDTypeOneSubStandard	Selects the ETSI FSK Caller ID Type 1 sub-standard (FXS only). 0 = ETSI between rings (default). 1 = ETSI before ring DT_AS. 2 = ETSI before ring RP_AS. 3 = ETSI before ring LR_DT_AS. 4 = ETSI not ring related DT_AS. 5 = ETSI not ring related RP_AS. 6 = ETSI not ring related LR_DT_AS.

Table 6-10: Analog Telephony Parameters (continues on pages 242 to 246)

ini File Field Name Web Parameter Name	Valid Range and Description
ETSIVMWITypeOneStandard	Selects the ETSI Visual Message Waiting Indication (VMWI) Type 1 sub-standard. 0 = ETSI VMWI between rings (default) 1 = ETSI VMWI before ring DT_AS 2 = ETSI VMWI before ring RP_AS 3 = ETSI VMWI before ring LR_DT_AS 4 = ETSI VMWI not ring related DT_AS 5 = ETSI VMWI not ring related RP_AS 6 = ETSI VMWI not ring related LR_DT_AS
BellcoreVMWITypeOneStandard	Selects the Bellcore VMWI sub-standard. 0 = Between rings (default). 1 = Not ring related.

6.5.11 Number Manipulation and Routing Parameters

Table 6-11: Number Manipulation and Routing Parameters (continues on pages 247 to 253)

***ini* File Field Name** **Web Parameter Name**	**Valid Range and Description**
TrunkGroup_x	▪ **For digital interfaces:** Defines the Trunk Group table. TrunkGroup_x = T-U/a-b,c,d x = Trunk group ID (1 to 99). T = Starting physical trunk number. U = Ending physical trunk number. a = Starting B-channel (from 1). b = Ending B-channel (up to 31). **Note:** To represent all B-channels use a single asterisk instead of 'a' and 'b'. c = Phone number associated with the first channel (optional). d = Optional Tel Profile ID (1 to 4). For example: TrunkGroup_1 = 0/1-31,1000 (for E1 span). TrunkGroup_1 = 1/1-31,$$,1. TrunkGroup_2 = 2/1-24,3000 (for T1 span). TrunkGroup_1 = 0-7/1-20,1000 (for 8 E1 spans, 20 B-channels). TrunkGroup_1 = 0-3/*,1000 (for 4 E1 spans, all B-channels). **Note:** An optional Tel Profile ID (1 to 9) can be applied to each group of B-channels. ▪ **For analog interfaces:** TrunkGroup_<Hunt Group ID> = <Starting channel> - <Ending channel>, <Phone Number>, <Tel Profile ID> For example: [Module 0] TrunkGroup_1 = 1-4,100 [Module 1] TrunkGroup_2 = 5-8,200,1 **Note 1:** The numbering of channels starts with 1. **Note 2:** 'Hunt Group ID' can be set to any number in the range 1 to 99. **Note 3:** This parameter can appear up to four times per each module. **Note 4:** An optional Tel ProfileID (1 to 9) can be applied to each group of channels.
ChannelList	This parameter is obsolete; use the parameter TrunkGroup_x instead.
DefaultNumber	For a description of this parameter, refer to Table 5-6 on page 80.
ChannelSelectMode	For a description of this parameter, refer to Table 5-3 on page 64.
TrunkGroupSettings [Trunk Group Settings]	Defines rules for port allocation for specific Trunk Groups, if such rule doesn't exist, the global rule defined by ChannelSelectMode applies. a, b, c a = Trunk Group ID number b = Channel select mode for that Trunk Group. c = Registration mode for that Trunk Group (Per Endpoint [0] or Per Gateway [1]). If not configured [-1], the value of AuthenticationMode is used. Available values are identical to those defined by the ChannelSelectMode parameter.
AddTrunkGroupAsPrefix	For a description of this parameter, refer to Table 5-14 on page 105.
AddPortAsPrefix	For a description of this parameter, refer to Table 5-14 on page 105.

Table 6-11: Number Manipulation and Routing Parameters (continues on pages 247 to 253)

ini File Field Name Web Parameter Name	Valid Range and Description
ReplaceEmptyDstWithPortNumber	For a description of this parameter, refer to Table 5-14 on page 105.
CopyDestOnEmptySource	0 = Leave Source Number empty (default). 1 = If the Source Number of an incoming Tel to IP call is empty, the Destination Number is copied to the Source Number.
AddNPlandTON2CallingNumber	For a description of this parameter, refer to Table 5-14 on page 105.
AddNPlandTON2CalledNumber	For a description of this parameter, refer to Table 5-14 on page 105.
UseSourceNumberAsDisplayName	For a description of this parameter, refer to Table 5-3 on page 64.
UseDisplayNameAsSourceNumber	For a description of this parameter, refer to Table 5-3 on page 64.
AlwaysUseRouteTable	For a description of this parameter, refer to Table 5-4 on page 72.
Prefix	Configures the Tel to IP Routing table to route incoming Tel calls to IP addresses. The format includes: Prefix = <Destination Phone Prefix>,<Destination IP Address>,<Source Phone Prefix>,<Profile ID> For example: Prefix = 20,10.2.10.2,202,1 Prefix = 10[340-451]xxx#,10.2.10.6,*,1 Prefix = *,gateway.domain.com,* **Note 1:** <destination / source phone prefix> can be single number or a range of numbers. **Note 2:** This parameter can appear up to 50 times. **Note 3:** Parameters can be skipped by using the sign '$$', for example: Prefix = $$,10.2.10.2,202,1 **Note 4:** The <Destination IP Address> field can be either in dotted format notation or a FQDN. This field can also include a selected port to use (in the format: <IP Address>:<Port>). If an FQDN is used, DNS resolution is performed according to DNSQueryType. **Note 5:** The IP address can include wildcards. The 'x' wildcard is used to represent single digits, e.g., 10.8.8.xx represents all addresses between 10.8.8.10 to 10.8.8.99. The '*' wildcard represents any number between 0 and 255, e.g., 10.8.8.* represents all addresses between 10.8.8.0 and 10.8.8.255. **Note 6:** If the string 'ENUM' is specified in the Destination IP Address field, an ENUM query containing the destination phone number is sent to the DNS server. The ENUM reply includes a SIP URI, used as the Request-URI in the outgoing INVITE and for routing (if Proxy is not used). **Note 7:** For configuring the Tel to IP Routing table using the Web interface, refer to Section 5.5.4.2 on page 107. For available notations, refer to Section 5.5.3.1 on page 102. For detailed information on this feature, refer to Section 5.5.4.2 on page 107.

Table 6-11: Number Manipulation and Routing Parameters (continues on pages 247 to 253)

ini File Field Name Web Parameter Name	Valid Range and Description
PSTNPrefix	PSTNPrefix = a,b,c,d,e a = Destination Number Prefix b = Trunk group ID (1 to 99) c = Source Number Prefix d = Source IP address (obtained from the Contact header in the INVITE message) e = IP Profile ID (1 to 4) Selection of trunk groups (for IP to Tel calls) is according to destination number, source number,and source IP address. **Note 1:** To support the 'in cal l alternative routing' feature, users can use two entries that support the same call, but assigned it with a different trunk groups. The second entree functions as an alternative selection if the first rule fails as a result of one of the release reasons listed in the AltRouteCauseIP2Tel table. **Note 2:** An optional IP ProfileID (1 to 4) can be applied to each routing rule. **Note 3:** The Source IP Address can include the 'x' wildcard to represent single digits. For example: 10.8.8.xx represents all IP addresses between 10.8.8.10 to 10.8.8.99. The '*' wildcard represents any number between 0 and 255, e.g., 10.8.8.* represents all addresses between 10.8.8.0 and 10.8.8.255. **Note 4:** If the Source IP field includes an FQDN, DNS resolution is performed according to DNSQueryType. **Note 5:** For available notations that represent multiple numbers, refer to Section 5.5.3.1 on page 102. **Note 6:** For configuring the IP to Trunk Group Routing table using the Web interface, refer to Section 5.5.4.3 on page 109.
RemovePrefix	For a description of this parameter, refer to Table 5-14 on page 105.
RouteModeIP2Tel	For a description of this parameter, refer to Table 5-16 on page 111.
RouteModeTel2IP	For a description of this parameter, refer to Table 5-15 on page 108.
SwapRedirectNumber	For a description of this parameter, refer to Table 5-22 on page 136.
Prefix2RedirectNumber	For a description of this parameter, refer to Table 5-22 on page 136.
AddTON2RPI	For a description of this parameter, refer to Table 5-3 on page 64.

Table 6-11: Number Manipulation and Routing Parameters (continues on pages 247 to 253)

ini File Field Name Web Parameter Name	Valid Range and Description
NumberMapTel2IP	Manipulates the destination number for Tel to IP calls (Destination Phone Number Manipulation Table for Tel→IP calls). NumberMapTel2IP = a,b,c,d,e,f,g a = Destination number prefix b = Number of stripped digits from the left, or (if brackets are used) from the right. A combination of both options is allowed. c = String to add as prefix, or (if brackets are used) as suffix. A combination of both options is allowed. d = Number of remaining digits from the right e = Number Plan used in RPID header f = Number Type used in RPID header g = Source number prefix The ‘b’ to ‘f’ manipulations rules are applied if the called and calling numbers match the ‘a’ and ‘g’ conditions. The manipulation rules are executed in the following order: ‘b’, ‘d’ and ‘c’. Parameters can be skipped by using the sign ‘$$’, for example: NumberMapTel2IP=01,2,972,$$,0,0,$$ NumberMapTel2IP=03,(2),667,$$,0,0,22 **Note:** Number Plan & Type can optionally be used in Remote Party ID (RPID) header by using the ‘EnableRPIHeader’ and ‘AddTON2RPI’ parameters.
NumberMapIP2Tel	Manipulate the destination number for IP to Tel calls (Destination Phone Number Manipulation Table for IP→Tel Calls). NumberMapIP2Tel = a,b,c,d,e,f,g,h,i a = Destination number prefix b = Number of stripped digits from the left, or (if brackets are used) from the right. A combination of both options is allowed. c = String to add as prefix, or (if brackets are used) as suffix. A combination of both options is allowed. d = Number of remaining digits from the right e = Q.931 Number Plan f = Q.931 Number Type g = Source number prefix h = Not applicable, set to $$ i = Source IP address (obtained from the Contact header in the INVITE message) The ‘b’ to ‘f’ manipulation rules are applied if the called and calling numbers match the ‘a’, ‘g’ and ‘i’ conditions. The manipulation rules are executed in the following order: ‘b’, ‘d’ and ‘c’. Parameters can be skipped by using the sign ‘$$’, for example: NumberMapIP2Tel =01,2,972,$$,0,$$,034 NumberMapIP2Tel =03,(2),667,$$,$$,0,22,$$,10.13.77.8 **Note:** The Source IP address can include the ‘x’ wildcard to represent single digits. For example: 10.8.8.xx represents all the addresses between 10.8.8.10 to 10.8.8.99. The ‘*’ wildcard represents any number between 0 and 255, e.g. 10.8.8.* represents all the addresses between 10.8.8.0 and 10.8.8.255.

Table 6-11: Number Manipulation and Routing Parameters (continues on pages 247 to 253)

***ini* File Field Name** **Web Parameter Name**	**Valid Range and Description**

SourceNumberMapTel2IP	Configures the Source Phone Number Manipulation Table for Tel→IP calls.
	SourceNumberMapTel2IP = a,b,c,d,e,f,g,h
	
	a = Source number prefix
	b = Number of stripped digits from the left, or (if in brackets are used) from right. A combination of both options is allowed.
	c = String to add as prefix, or (if in brackets are used) as suffix. A combination of both options is allowed.
	d = Number of remaining digits from the right
	e = Number Plan used in RPID header
	f = Number Type used in RPID header
	g =Destination number prefix
	h =Calling number presentation (0 to allow presentation, 1 to restrict presentation)
	
	The 'b' to 'f' and 'h' manipulation rules are applied if the called and calling numbers match the 'a' and 'g' conditions.
	
	The manipulation rules are executed in the following order: 'b', 'd' and 'c'. Parameters can be skipped by using the sign '$$', for example:
	SourceNumberMapTel2IP=01,2,972,$$,0,0,$$,1
	SourceNumberMapTel2IP=03,(2),667,$$,0,0,22,0
	Note 1: 'Presentation' is set to 'Restricted' only if 'Asserted Identity Mode' is set to 'P-Asserted'.
	Note 2: Number Plan & Type can optionally be used in Remote Party ID (RPID) header by using the 'EnableRPIHeader' and 'AddTON2RPI' parameters.

SourceNumberMapIP2Tel	Manipulate the source number for IP to Tel calls (Source Phone Number Manipulation Table for IP→Tel calls).
	SourceNumberMapIP2Tel = a,b,c,d,e,f,g,h,i
	
	a = Source number prefix
	b = Number of stripped digits from the left, or (if brackets are used) from the right. A combination of both options is allowed.
	c = String to add as prefix, or (if brackets are used) as suffix. A combination of both options is allowed.
	d = Number of remaining digits from the right
	e = Q.931 Number Plan
	f = Q.931 Number Type
	g = Destination number prefix
	h =Calling number presentation (0 to allow presentation, 1 to restrict presentation)
	I = Source IP address (obtained from the Request-URI in the INVITE message).
	
	The 'b' to 'f' and 'h' manipulation rules are applied if the called and calling numbers match the 'a', 'g' and 'I' conditions.
	
	The manipulation rules are executed in the following order: 'b', 'd' and 'c'. Parameters can be skipped by using the sign '$$', for example:
	SourceNumberMapIP2Tel =01,2,972,$$,0,$$,034,1
	SourceNumberMapIP2Tel =03,(2),667,$$,$$,0,22
	Note: The Source IP address can include the 'x' wildcard to represent single digits. For example: 10.8.8.xx represents all the addresses between 10.8.8.10 to 10.8.8.99. The '*' wildcard represents any number between 0 and 255, e.g. 10.8.8.* represents all the addresses between 10.8.8.0 and 10.8.8.255.

Table 6-11: Number Manipulation and Routing Parameters (continues on pages 247 to 253)

ini File Field Name Web Parameter Name	Valid Range and Description
For ETSI ISDN variant, the following Number Plan and Type combinations (Plan/Type) are supported in the Destination and Source Manipulation tables: 0,0 = Unknown, Unknown 9,0 = Private, Unknown 9,1 = Private, Level 2 Regional 9,2 = Private, Level 1 Regional 9,3 = Private, PISN Specific 9,4 = Private, Level 0 Regional (local) 1,0 = Public(ISDN/E.164), Unknown 1,1 = Public(ISDN/E.164), International 1,2 = Public(ISDN/E.164), National 1,3 = Public(ISDN/E.164), Network Specific 1,4 = Public(ISDN/E.164), Subscriber 1,6 = Public(ISDN/E.164), Abbreviated For NI-2 and DMS-100 ISDN variants the valid combinations of TON and NPI for calling and called numbers are (Plan/Type): 0/0 - Unknown/Unknown 1/1 - International number in ISDN/Telephony numbering plan 1/2 - National number in ISDN/Telephony numbering plan 1/4 - Subscriber (local) number in ISDN/Telephony numbering plan 9/4 - Subscriber (local) number in Private numbering plan	
SecureCallsFromIP	For a description of this parameter, refer to Table 5-7 on page 85.
AltRouteCauseTel2IP	Table of call failure reason values received from the IP side. If a call is released as a result of one of these reasons, the gateway tries to find an alternative route to that call in the 'Tel to IP Routing' table. For example: AltRouteCauseTel2IP = 486 (Busy here). AltRouteCauseTel2IP = 480 (Temporarily unavailable). AltRouteCauseTel2IP = 408 (No response). **Note 1:** The 408 reason can be used to specify that there was no response from the remote party to the INVITE request. **Note 2:** This parameter can appear up to 5 times. **Note 3:** For defining the Reasons for Alternative Routing table using the Web interface, refer to Section 5.5.4.6 on page 113.
AltRouteCauseIP2Tel	Table of call failure reason values received from the PSTN side (in Q.931 presentation). If a call is released as a result of one of these reasons, the gateway tries to find an alternative trunk group to that call in the 'IP to Trunk Group Routing' table. For example: AltRouteCauseIP2Tel = 3 (No route to destination). AltRouteCauseIP2Tel = 1 (Unallocated number). AltRouteCauseIP2Tel = 17 (Busy here). **Note 1:** This parameter can appear up to 5 times. **Note 2**: If the gateway fails to establish a cal to the PSTN because it has no available channels in a specific trunk group (e.g., all of the trunk group's channels are occupied, or the trunk group's spans are disconnected or out of sync), it uses the internal release cause '3' (no route to destination). This cause can be used in the 'AltRouteCauseIP2Tel' table to define routing to an alternative trunk group. **Note 3:** For defining the Reasons for Alternative Routing table using the Web interface, refer to Section 5.5.4.6 on page 113.
FilterCalls2IP	For a description of this parameter, refer to Table 5-7 on page 85.

Table 6-11: Number Manipulation and Routing Parameters (continues on pages 247 to 253)

ini File Field Name Web Parameter Name	Valid Range and Description
Alternative Routing Parameters	
AltRoutingTel2IPEnable	For a description of this parameter, refer to Table 5-14 on page 105.
AltRoutingTel2IPMode	For a description of this parameter, refer to Table 5-14 on page 105.
IPConnQoSMaxAllowedPL	For a description of this parameter, refer to Table 5-14 on page 105.
IPConnQoSMaxAllowedDelay	For a description of this parameter, refer to Table 5-14 on page 105.
Phone-Context Parameters	
AddPhoneContextAsPrefix	For a description of this parameter, refer to Table 5-13 on page 104.
PhoneContext	Defines the Phone Context table. When a call is received from the ISDN, the NPI and TON are compared against the table, and the Phone-Context value is used in the outgoing SIP INVITE message. The same mapping occurs when an INVITE with a Phone-Context attribute is received. The Phone-Context parameter appears in the standard SIP headers where a phone number is used (Request-URI, To, From, Diversion). PhoneContext = <Number Plan>,<Number Type>,<Phone-Context> For example: ▪ PhoneContext = 0,0,unknown.com ▪ PhoneContext = 1,1,host.com ▪ PhoneContext = 9,1,na.e164.host.com **Note 1:** This parameter can appear up to 20 times. **Note 2:** Several rows with the same NPI-TON or Phone-Context are allowed. In this scenario, a Tel-to-IP call uses the first match. **Note 3:** Phone-Context '+' is a unique case as it doesn't appear in the Request-URI as a Phone-Context parameter. Instead, it's added as a prefix to the phone number. The '+' isn't removed from the phone number in the IP-to-Tel direction. **Note 4:** To configure Phone Context table using the Web interface, refer to Section 5.5.3.3 on page 103.

6.5.12 E1/T1 Configuration Parameters

Table 6-12: E1/T1/J1 Configuration Parameters (continues on pages 253 to 255)

ini File Field Name Web Parameter Name	Valid Range and Description
PCMLawSelect	For a description of this parameter, refer to Table 5-34 on page 166.
ProtocolType	For a description of this parameter, refer to Table 5-33 on page 162.
ProtocolType_x	Same as the description for parameter ProtocolType, but for a specific trunk ID.
TraceLevel	For a description of this parameter, refer to Table 5-33 on page 162.
FramingMethod	For a description of this parameter, refer to Table 5-33 on page 162.
FramingMethod_x	Same as the description for parameter FramingMethod, but for a specific trunk ID.
TerminationSide	For a description of this parameter, refer to Table 5-33 on page 162.

Table 6-12: E1/T1/J1 Configuration Parameters (continues on pages 253 to 255)

ini File Field Name Web Parameter Name	Valid Range and Description
TerminationSide_x	Same as the description for parameter TerminationSide, but for a specific trunk ID.
ClockMaster	For a description of this parameter, refer to Table 5-33 on page 162.
ClockMaster_x	Same as the description for parameter ClockMaster, but for a specific trunk ID.
TDMBusClockSource	For a description of this parameter, refer to Table 5-34 on page 166.
TDMBusPSTNAutoClockEnable	For a description of this parameter, refer to Table 5-34 on page 166.
TDMBusLocalReference	For a description of this parameter, refer to Table 5-34 on page 166.
AutoClockTrunkPriority	For a description of this parameter, refer to Table 5-33 on page 162.
TDMBusPSTNAutoClockRevertingEnable	For a description of this parameter, refer to Table 5-34 on page 166.
LineCode	For a description of this parameter, refer to Table 5-33 on page 162.
LineCode_x	Same as the description for parameter LineCode, but for a specific trunk ID.
BChannelNegotiation	For a description of this parameter, refer to Table 5-22 on page 136.
NFASGroupNumber_x	For a description of this parameter, refer to Table 5-33 on page 162.
DChConfig_x	For a description of this parameter, refer to Table 5-33 on page 162.
ISDNNFASInterfaceID_x	For a description of this parameter, refer to Table 5-33 on page 162.
CASTableIndex_x	For a description of this parameter, refer to Table 5-33 on page 162.
CASFileName_0 **CASFileName_1** **CASFileName_7**	CAS file name (e.g., 'E_M_WinkTable.dat') that defines the CAS protocol. It is possible to define up to eight different CAS files by repeating this parameter. Each CAS file can be associated with one or more of the gateway trunks using the parameter CASTableIndex_x.
CASTablesNum	1 to 8. Indicates how many CAS protocol configurations files are loaded.
IdleABCDPattern	For a description of this parameter, refer to Table 5-34 on page 166.
IdlePCMPattern	For a description of this parameter, refer to Table 5-34 on page 166.
LineBuildOut.Loss	For a description of this parameter, refer to Table 5-33 on page 162.
ISDNRxOverlap_x	For a description of this parameter, refer to Table 5-33 on page 162.
ISDNRxOverlap	0 = Disabled (default). 1 = Enabled. Any number bigger than one = Number of digits to receive. **Note 1:** If enabled the gateway receives ISDN called number that is sent in the 'Overlap' mode. **Note 2:** The INVITE to IP is sent only after the number (including 'Sending Complete' Info Element) was fully received (in SETUP and/or subsequent INFO Q.931 messages). For detailed information on ISDN overlap dialing, refer to Section 11.1 on page 315.
SendMetering2IP	For a description of this parameter, refer to Table 5-18 on page 127.
DigitMapping	For a description of this parameter, refer to Table 5-6 on page 80.
TimeBetweenDigits	For a description of this parameter, refer to Table 5-6 on page 80.
MaxDigits	For a description of this parameter, refer to Table 5-6 on page 80.
TimeForDialTone	For a description of this parameter, refer to Table 5-6 on page 80.
R2Category	For a description of this parameter, refer to Table 5-22 on page 136.
RegretTime	For a description of this parameter, refer to Table 5-7 on page 85.

Table 6-12: E1/T1/J1 Configuration Parameters (continues on pages 253 to 255)

ini File Field Name Web Parameter Name	Valid Range and Description
HeldTimeout	Determines the time period the gateway can stay on-hold. If a Resume (un-hold Re- INVITE) message is received before the timer expires, the call is renewed. If this timer expires, the call is released. -1 = Indefinitely (default). 0 - 2400 =Time to wait in seconds. Currently applicable only to MFC R2 CAS variants.
TrunkLifeLineType	Defines the type of trunk lifeline. Short trunks 1-2, 3-4. 0 = Activate lifeline on power down (default) 1 = Activate lifeline on power down or on detection of LAN disconnect 2 = Activate lifeline on power down or on detection of LAN disconnect or loss of ping
TrunkAdministrativeState	Defines the administrative state of a trunk. 0 = Lock the trunk; stops trunk traffic to configure the trunk protocol type 2 = Unlock the trunk (default); enables trunk traffic
ISDN Flexible Behavior Parameters ISDN protocol is implemented in different Switches / PBXs by different vendors. Several implementations vary a little from the specification. Therefore, to provide a flexible interface that supports these ISDN variants, the ISDN behavior parameters are used.	
ISDNInCallsBehavior	For a description of this parameter, refer to Table 5-33 on page 162.
ISDNIBehavior	For a description of this parameter, refer to Table 5-33 on page 162.
ISDNGeneralCCBehavior	For a description of this parameter, refer to Table 5-33 on page 162.
ISDNOutCallsBehavior	For a description of this parameter, refer to Table 5-33 on page 162.
ISDNIBehavior_x	Same as the description for parameter ISDNIBehavior, but for a specific trunk ID.
ISDNInCallsBehavior_x	Same as the description for parameter ISDNInCallsBehavior, for a specific trunk ID.
ISDNOutCallsBehavior_x	Same as the description for parameter ISDNOutCallsBehavior, but for a specific trunk ID.

6.5.13 Channel Parameters

The Channel Parameters define the DTMF, fax and modem transfer modes.

Table 6-13: Channel Parameters (continues on pages 255 to 258)

ini File Field Name Web Parameter Name	Valid Range and Description
DJBufMinDelay	For a description of this parameter, refer to Table 5-30 on page 155.
DJBufOptFactor	For a description of this parameter, refer to Table 5-30 on page 155.
FaxTransportMode	For a description of this parameter, refer to Table 5-29 on page 152.
FaxRelayEnhancedRedundancyDepth	For a description of this parameter, refer to Table 5-29 on page 152.
FaxRelayRedundancyDepth	For a description of this parameter, refer to Table 5-29 on page 152.
FaxRelayMaxRate	For a description of this parameter, refer to Table 5-29 on page 152.
FaxRelayECMEnable	For a description of this parameter, refer to Table 5-29 on page 152.
FaxModemBypassCoderType	For a description of this parameter, refer to Table 5-29 on page 152.
CNGDetectorMode	For a description of this parameter, refer to Table 5-29 on page 152.

Table 6-13: Channel Parameters (continues on pages 255 to 258)

ini File Field Name Web Parameter Name	Valid Range and Description
FaxModemBypassM	For a description of this parameter, refer to Table 5-29 on page 152.
FaxBypassPayloadType	For a description of this parameter, refer to Table 5-30 on page 155.
ModemBypassPayloadType	Modem Bypass dynamic payload type (range 0-127). The default value is 103.
FaxModemRelayVolume	-18 to -3, corresponding to -18 dBm to -3 dBm in 1 dB steps. (Default = -12 dBm) fax gain control.
DetFaxOnAnswerTone	For a description of this parameter, refer to Table 5-3 on page 64.
FaxModemBypassBasicRTPPacketInterval	Determines the basic frame size that is used during fax / modem bypass sessions. 0 = set internally (default) 1 = 5 msec (not recommended) 2 = 10 msec 3 = 20 msec **Note:** When set for 5 msec (1), the maximum number of simultaneous channels supported is 120.
FaxModemBypassDJBufMinDelay	0 to 150 msec (default=40) Determines the Jitter Buffer delay during fax and modem bypass session
NSEMode	Cisco compatible fax and modem bypass mode 0 = NSE disabled (default) 1 = NSE enabled **Note 1:** This feature can be used only if VxxModemTransportType=2 (Bypass) **Note 2:** If NSE mode is enabled the SDP contains the following line: 'a=rtpmap:100 X-NSE/8000' **Note 3:** To use this feature: ▪ The Cisco gateway must include the following definition: 'modem passthrough nse payload-type 100 codec g711alaw'. ▪ Set the Modem transport type to Bypass mode ('VxxModemTransportType = 2') for all modems. ▪ Configure the gateway parameter NSEPayloadType= 100 In NSE bypass mode the gateway starts using G.711 A-Law (default) or G.711μ-Law, according to the parameter 'FaxModemBypassCoderType'. The payload type used with these G.711 coders is a standard one (8 for G.711 A-Law and 0 for G.711 μ-Law). The parameters defining payload type for the 'old' AudioCodes' Bypass mode. 'FaxBypassPayloadType' and 'ModemBypassPayloadType' are not used with NSE Bypass. The bypass packet interval is selected according to the parameter 'FaxModemBypassBasicRtpPacketInterval'.
NSEPayloadType	NSE payload type for Cisco Bypass compatible mode. The valid range is 96-127. The default value is 105. **Note:** Cisco gateways usually use NSE payload type of 100.
V21ModemTransportType	For a description of this parameter, refer to Table 5-29 on page 152.
V22ModemTransportType	For a description of this parameter, refer to Table 5-29 on page 152.
V23ModemTransportType	For a description of this parameter, refer to Table 5-29 on page 152.
V32ModemTransportType	For a description of this parameter, refer to Table 5-29 on page 152.
V34ModemTransportType	For a description of this parameter, refer to Table 5-29 on page 152.
BellModemTransportType	Determines the Bell modem transport method. 0 = Transparent (default). 2 = Bypass. 3 = Transparent with events.
InputGain	For a description of this parameter, refer to Table 5-28 on page 149.

Table 6-13: Channel Parameters (continues on pages 255 to 258)

ini File Field Name Web Parameter Name	Valid Range and Description
VoiceVolume	For a description of this parameter, refer to Table 5-28 on page 149.
RTPRedundancyDepth	For a description of this parameter, refer to Table 5-30 on page 155.
RFC2198PayloadType	For a description of this parameter, refer to Table 5-30 on page 155.
EnableSilenceCompression	For a description of this parameter, refer to Table 5-28 on page 149.
IsCiscoSCEMode	0 = There isn't a Cisco gateway at the remote side (default). 1 = There is a Cisco gateway at the remote side. When there is a Cisco gateway at the remote side, the local gateway must set the value of the 'annexb' parameter of the fmtp attribute in the SDP to 'no'. This logic should be used if 'EnableSilenceCompression = 2' (enable without adaptation). In this case, Silence Suppression should be used on the channel but not declared in the SDP.
EnableEchoCanceller	For a description of this parameter, refer to Table 5-28 on page 149.
MaxEchoCancellerLength	For a description of this parameter, refer to Table 5-32 on page 159.
EchoCancellerAggressiveNLP	Enables or disables the Aggressive Non-Linear Processor (NLP) in the first 0.5 second of the call. 0 = Disabled (default) 1 = Enabled
EnableNoiseReduction	Enables / disables the DSP Noise Reduction mechanism. 0 = Disable (default). 1 = Enable. **Note:** When this parameter is enabled the channel capacity might be reduced.
TestMode	0 = CoderLoopback, encoder-decoder loopback inside DSP. 1 = PCMLoopback, loopback the incoming PCM to the outgoing PCM. 2 = ToneInjection, generates a 1000 Hz tone to outgoing PCM. 3 = NoLoopback, (default).
EnableStandardSIDPayloadType	For a description of this parameter, refer to Table 5-30 on page 155.
ComfortNoiseNegotiation	For a description of this parameter, refer to Table 5-30 on page 155.
RTPSIDCoeffNum	Determines the number of spectral coefficients added to an SID packet being sent according to RFC 3389. Valid only if 'EnableStandardSIDPayloadType' is set to 1. The valid values are 0 (default), 4, 6, 8 and 10.
DTMFVolume	For a description of this parameter, refer to Table 5-28 on page 149.
RxDTMFHangOverTime	Defines the Voice Silence time (in msec units) after playing DTMF or MF digits to the Tel / PSTN side that arrive as Relay from the IP side. Valid range is 0 to 2,000 msec. Default is 1,000 msec.
TxDTMFHangOverTime	Defines the Voice Silence time (in msec units) after detecting the end of DTMF or MF digits at the Tel / PSTN side when the DTMF Transport Type is either Relay or Mute. Valid range is 0 to 2,000 msec. Default is 100 msec.
DTMFTransportType	For a description of this parameter, refer to Table 5-28 on page 149.
AnswerDetectorSensitivit	For a description of this parameter, refer to Table 5-28 on page 149.
RFC2833PayloadType	For a description of this parameter, refer to Table 5-6 on page 80.
MGCPDTMFDetectionPoint	0 = DTMF event is reported on the end of a detected DTMF digit. 1 = DTMF event is reported on the start of a detected DTMF digit (default).
DTMFInterDigitInterval	Time in msec between generated DTMF digits to PSTN side (if TxDTMFOption = 1, 2 or 3). The default value is 100 msec. The valid range is 0 to 32767.

Table 6-13: Channel Parameters (continues on pages 255 to 258)

ini File Field Name Web Parameter Name	Valid Range and Description
DTMFDigitLength	Time in msec for generating DTMF tones to the PSTN side (if TxDTMFOption = 1, 2 or 3). The default value is 100 msec. The valid range is 0 to 32767.
AnalogSignalTransportType	For a description of this parameter, refer to Table 5-30 on page 155.
VQMonEnable	For a description of this parameter, refer to Table 5-30 on page 155.
RTCPInterval	For a description of this parameter, refer to Table 5-30 on page 155.
DisableRTCPRandomize	For a description of this parameter, refer to Table 5-30 on page 155.
RTCPXREscIP	For a description of this parameter, refer to Table 5-30 on page 155.
RTCPXRReportMode	For a description of this parameter, refer to Table 5-30 on page 155.
MinFlashHookTime	For a description of this parameter, refer to Table 5-31 on page 158.
FlashHookPeriod	For a description of this parameter, refer to Table 5-31 on page 158.

6.5.14 Configuration Files Parameters

The configuration files (Call Progress Tones, Prerecorded Tones, User Information, FXS/FXO Coefficient, and CAS) can be loaded to the Mediant 1000 using the Embedded Web Server or a TFTP session (refer to Section 5.8.2 on page 196). Before you load them to the gateway, in the *ini* file, you need to specify the files that you want loaded and whether they must be stored in the non-volatile memory. The table below lists the *ini* file parameters associated with these configuration files:

Table 6-14: Configuration Files Parameters

ini File Field Name	Valid Range and Description
CallProgressTonesFilename	The name of the file containing the Call Progress Tones definitions. Refer to Section 17.1 on page 391 for additional information on how to create and load this file.
FXSLoopCharacteristicsFileName	The name (and path) of the file providing the FXS line characteristic parameters.
FXOLoopCharacteristicsFileName	The name (and path) of the file providing the FXO line characteristic parameters.
CASFileName	This is the name of the file containing specific CAS protocol definition (such as 'E_M_WinkTable.dat'). These files are provided to support various types of CAS signaling.
CASFileName_x	CAS file name (e.g., 'E_M_WinkTable.dat') that defines the CAS protocol. It is possible to define up to eight different CAS files by repeating this parameter. Each CAS file can be associated with one or more of the gateway trunks using the parameter CASTableIndex_x.
CASTablesNum	Number, 1 to 8. Specifies how many CAS configuration files are loaded.
PrerecordedTonesFileName	The name (and path) of the file containing the Prerecorded Tones.
UserInfoFileName	The name (and path) of the file containing the User Information data.
SaveConfiguration	Determines if the gateway's configuration (parameters and files) is saved to flash (non-volatile memory). 0 = Configuration isn't saved to flash memory. 1 = Configuration is saved to flash memory (default).

7 Using BootP / DHCP

The Mediant 1000 uses the Bootstrap Protocol (BootP) and the Dynamic Host Configuration Protocol (DHCP) to obtain its networking parameters and configuration automatically after it is reset. BootP and DHCP are also used to provide the IP address of a TFTP server on the network, and files (*cmp* and *ini*) to be loaded into memory.

DHCP is a communication protocol that automatically assigns IP addresses from a central point. BootP is a protocol that enables a device to discover its own IP address. Both protocols have been extended to enable the configuration of additional parameters specific to the Mediant 1000.

Note: BootP is normally used to initially configure the Mediant 1000. Thereafter, BootP is no longer required as all parameters can be stored in the gateway’s non-volatile memory and used when BootP is inaccessible. BootP can be used again to change the IP address of the Mediant 1000 (for example).

7.1 BootP/DHCP Server Parameters

BootP/DHCP can be used to provision the following parameters (included in the BootP/DHCP reply). Note that only the IP address and subnet mask are mandatory:

- IP address, subnet mask - These mandatory parameters are sent to the Mediant 1000 every time a BootP/DHCP process occurs.
- Default gateway IP address - An optional parameter that is sent to the Mediant 1000 only if configured in the BootP/DHCP server.
- TFTP server IP address - An optional parameter that contains the address of the TFTP server from which the firmware (*cmp*) and ini files are loaded.
- DNS server IP address (primary and secondary) - Optional parameters that contain the IP addresses of the primary and secondary DNS servers. These parameters are available only in DHCP and from Boot version 1.92.
- Syslog server IP address - An optional parameter that is sent to the Mediant 1000 only if configured. This parameter is available only in DHCP.
- SIP server IP address – Two optional parameters that are sent to the Mediant 1000 only if configured. These parameters are available only in DHCP.
- Firmware file name – An optional parameter that contains the name of the firmware file to be loaded to the gateway via TFTP.
- *ini* file name - An optional parameter that contains the name of the *ini* file to be loaded to the gateway via TFTP.

7.2 Using DHCP

When the gateway is configured to use DHCP (DHCPEnable = 1), it attempts to contact the local DHCP server to obtain the networking parameters (IP address, subnet mask, default gateway, primary/secondary DNS server and two SIP server addresses). These network parameters have a ‘time limit’. After the time limit expires, the gateway must ‘renew’ its lease from the DHCP server.

Note that if the DHCP server denies the use of the gateway's current IP address and specifies a different IP address (according to RFC 1541), the gateway must change its networking parameters. If this happens while calls are in progress, they are not automatically rerouted to the new network address (since this function is beyond the scope of a VoIP gateway). Therefore, administrators are advised to configure DHCP servers to allow renewal of IP addresses.

Notes:

- If, during operation, the IP address of the gateway is changed as a result of a DHCP renewal, the gateway is automatically reset.
- If the gateway's network cable is disconnected and reconnected, a DHCP renewal is performed (to verify that the gateway is still connected to the same network).
  When DHCP is enabled, the gateway also includes its product name (e.g., 'Mediant 1000') in the DHCP 'option 60' Vendor Class Identifier. The DHCP server can use this product name to assign an IP address accordingly.
- After power-up, the gateway performs two distinct DHCP sequences. Only in the second sequence, DHCP 'option 60' is contained. If the gateway is reset from the Web/SNMP, only a single DHCP sequence containing 'option 60' is sent. If DHCP procedure is used, the new gateway IP address, allocated by the DHCP server, must be detected

➢ **To detect the gateway's IP address, follow one of the procedures below:**

- Starting with Boot version 1.92, the gateway can use a host name in the DHCP request. The host name is set to acl_nnnnn, where nnnnn stands for the gateway's serial number (the serial number is equal to the last 6 digits of the MAC address converted from Hex to decimal). If the DHCP server registers this host name to a DNS server, the user can access the gateway (through a Web browser) using a URL of http://acl_<serial number> (instead of using the gateway's IP address). For example, if the gateway's MAC address is 00908f010280, the DNS name is acl_66176.
- After physically resetting the gateway its IP address is displayed in the 'Client Info' column in the BootP/TFTP configuration utility (refer to Figure D-1 on page 415).
- Use a serial communication software (refer to Section 4.2.4 on page 50).
- Contact your System Administrator.

7.3 Using BootP

7.3.1 Upgrading the Mediant 1000

When upgrading the Mediant 1000 (loading new software onto the gateway) using the BootP/TFTP configuration utility:

- From version 4.4 to version 4.4 or to any higher version, the device retains its configuration (*ini* file). However, the auxiliary files (CPT, logo, etc.) may be erased.
- From version 4.6 to version 4.6 or to any higher version, the device retains its configuration (*ini* file) and auxiliary files (CPT, logo, etc.).

You can also use the Software Upgrade wizard, available through the Web Interface (refer to Section 5.8.1 on page 191).

Note: To save the *cmp* file to non-volatile memory, use the **-fb** command line switch. If the file is not saved, the gateway reverts to the old version of software after the next reset. For information on using command line switches, refer to Section D.11.6 on page 422.

7.3.2 Vendor Specific Information Field

The Mediant 1000 uses the vendor specific information field in the BootP request to provide device-related initial startup information. The BootP/TFTP configuration utility displays this information in the ‘Client Info’ column (refer to Figure D-1).

Note: This option is not available on DHCP servers.

The Vendor Specific Information field is disabled by default. To enable / disable this feature: set the *ini* file parameter ExtBootPReqEnable (Section 6.5.2 on page 220) or use the **-be** command line switch (refer to Table D-1 on page 423).

Table 7-1 details the vendor specific information field according to device types:

Table 7-1: Vendor Specific Information Field

Tag #	Description	Value	Length
220	Gateway Type	#9 = Mediant 1000	1
221	Current IP Address	XXX.XXX.XXX.XXX	4
222	Burned Boot Software Version	X.XX	4
223	Burned *cmp* Software Version	XXXXXXXXXXXX	12
224	Geographical Address	0 – 31	1
225	Chassis Geographical Address	0 – 31	1
228	Indoor / Outdoor (Indoor is valid only for FXS. FXO is always Outdoor.)	#0 = Indoor #1 = Outdoor	1
229	E&M	N/A	1
230	Analog Channels	4 / 8 / 12 / 16 / 20 / 24	1

Table 7-2 exemplifies the structure of the vendor specific information field for a TP-1610 slave module with IP address 10.2.70.1.

Table 7-2: Structure of the Vendor Specific Information Field

Vendor-Specific Information Code	Length Total	Tag Num	Length	Value	Tab Num	Length	Value	Tag Num	Length	Value (1)	Value (2)	Value (3)	Value (4)	Tag End
42	**12**	**220**	**1**	**2**	**225**	**1**	**1**	**221**	**4**	**10**	**2**	**70**	**1**	**255**

Reader's Notes

8 Media Server Capabilities

This chapter provides information on the Mediant 1000 conferencing capabilities.

1.1 Conference Server -- Simple Conferencing (NetAnn)

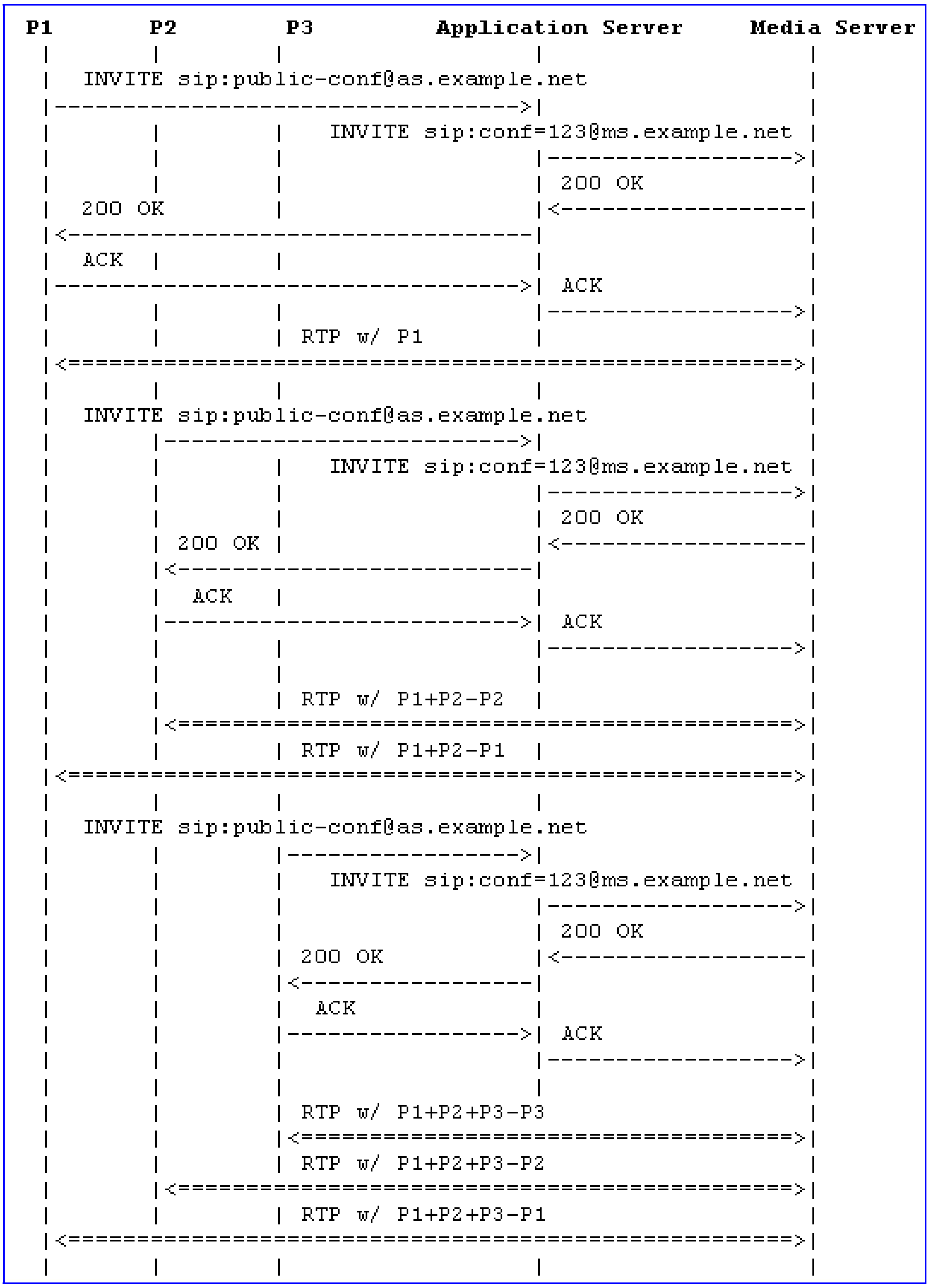
Note: The conference application is a special order option.

The Mediant 1000 server supports dial-in, multi-party conferencing. In conference applications, the Mediant 1000 server functions as a centralized conference bridge. In ad-hoc or prearranged conferences, users ‘invite’ the conference bridge. The conference bridge combines the media and sends it to all participants.

The Mediant 1000 supports the Simple conferencing interface (according to NetAnn).

1.1.1 SIP Call Flow

Figure 8-1: Simple Conferencing SIP Call Flow



```
P1       P2        P3          Application Server     Media Server
 |       |        |                  |                   |
 |  INVITE sip:public-conf@as.example.net                |
 |----------------------------------->|                   |
 |       |        |    INVITE sip:conf=123@ms.example.net |
 |       |        |                  |------------------>|
 |       |        |                  | 200 OK            |
 |  200 OK        |                  |<------------------|
 |<-----------------------------------|                   |
 |  ACK  |        |                  |                   |
 |----------------------------------->| ACK               |
 |       |        |                  |------------------>|
 |       |        | RTP w/ P1        |                   |
 |<=====================================================>|
 |       |        |                  |                   |
 |  INVITE sip:public-conf@as.example.net                |
 |       |--------------------------->|                   |
 |       |        |    INVITE sip:conf=123@ms.example.net |
 |       |        |                  |------------------>|
 |       |        |                  | 200 OK            |
 |       | 200 OK |                  |<------------------|
 |       |<---------------------------|                   |
 |       |  ACK   |                  |                   |
 |       |--------------------------->| ACK               |
 |       |        |                  |------------------>|
 |       |        |                  |                   |
 |       |        | RTP w/ P1+P2-P2  |                   |
 |       |<=============================================>|
 |       |        | RTP w/ P1+P2-P1  |                   |
 |<=====================================================>|
 |       |        |                  |                   |
 |  INVITE sip:public-conf@as.example.net                |
 |       |        |------------------>|                   |
 |       |        |    INVITE sip:conf=123@ms.example.net |
 |       |        |                  |------------------>|
 |       |        |                  | 200 OK            |
 |       |        | 200 OK           |<------------------|
 |       |        |<------------------|                   |
 |       |        |  ACK             |                   |
 |       |        |------------------>| ACK               |
 |       |        |                  |------------------>|
 |       |        |                  |                   |
 |       |        | RTP w/ P1+P2+P3-P3                   |
 |       |        |<====================================>|
 |       |        | RTP w/ P1+P2+P3-P2                   |
 |       |<=============================================>|
 |       |        | RTP w/ P1+P2+P3-P1                   |
 |<=====================================================>|
 |       |        |                  |                   |
```

1.1.2 Creating a Conference

The Mediant 1000 conference server creates a conference call when the first user joins the conference. To create a conference, the Application Server should send a regular SIP INVITE message to the Mediant 1000. The User Part of that Request-URI should include both the Conference Service Identifier (indicating that the requested Media Service is a Conference) and a Unique Conference Identifier (identifying a specific instance of a conference).

```
INVITE sip: conf100@audiocodes.com SIP/2.0
```

By default, a request to create a conference reserves three resources on the Mediant 1000. It is possible to reserve a larger number of resources in advance by adding the number of required participants to the User Part of the Request-URI. For example, '6conf100' reserves six resources for the duration of the conference.

If the Mediant 1000 can allocate the requested number of resources, it responds with a 200 OK.

The Conference Service Identifier can be set via the Mediant 1000 *ini* file (ConferenceID) or via the Web interface. By default, it is set to 'conf'.

1.1.3 Joining a Conference

To join an existing conference, the Application Server sends a SIP INVITE message with the same Request-URI as the one that created the conference. Each conference participant can use a different coder, negotiated with the Mediant 1000 server using usual SIP negotiation.

If more than the initially requested number of participants try to join the conference (i.e., four resources were reserved and a fifth INVITE is received) and the Mediant 1000 has an available resource, that request shall be granted.

If an INVITE to join an existing conference is received with a request to reserve a larger number of participants than initially requested, it shall be granted if the Mediant 1000 has enough available resources. A request for a smaller number of participants shall not be granted as this might create a situation that existing legs need to be disconnected.

The maximal number of participants in a single conference is 60. The maximal number of participants that actually participate in the mix at a given time is 3 (the loudest legs).

The Application Server can place a participant on Hold/Un-hold by sending the appropriate SIP Re-INVITE on that participant dialog.

1.1.4 Terminating a Conference

The Mediant 1000 never disconnects an existing conference leg. If a BYE is received on an existing leg, it is disconnected, but the resource is still saved if the same leg (or a different one) wants to re-join the conference. This logic occurs only for the initial number of reserved legs.

For example:

1. INVITE reserves three legs.
2. A, B, and C joins.
3. A disconnects.
4. A joins (guaranteed).
5. D joins.
6. A disconnects.
7. A joins (not guaranteed).

Sending a BYE request to the Mediant 1000 server terminates the participant's SIP session and removes it from the conference. The final BYE from the last participant ends the conference and releases all conference resources.

1.1.5 PSTN Participants

Adding PSTN participants is done by performing loopback from the IP side (TEL2IP have the Mediant 1000 IP address).

If the destination phone number in the incoming call from the PSTN is equal to the Conference Service Identifier and Unique Conference Identifier, the participant joins the conference.

A PSTN participant uses two DSP channels (caused by the IP loopback).

9 Telephony Capabilities

9.1 Working with Supplementary Services

The Mediant 1000 gateway supports the following supplementary services:

- Call Hold and Retrieve; refer to Section 9.1.1.
- Consultation / Alternate; refer to Section 9.1.2.
- Call Transfer (Refer + Replaces); refer to Section 9.1.3.
- Call Forward (3xx Redirect Responses); refer to Section 9.1.4.
- Call Waiting (182 Queued Response); refer to Section 9.1.5.
- Message Waiting Indication (MWI); refer to Section 9.1.6.

To activate these supplementary services (Hold, Transfer, Forward, Waiting and MWI) on the Mediant 1000 gateway, enable each service's corresponding parameter either from the Web Interface or via the *ini* file. Note that all call participants must support the specific used method.

Note: When working with application servers (such as BroadSoft's BroadWorks) in client server mode (the application server controls all supplementary services and keypad features by itself), the gateway's supplementary services must be disabled.

9.1.1 Call Hold and Retrieve

9.1.1.1 Initiating Hold/Retrieve

- Active calls can be put on-hold by pressing the phone's hook-flash button.
- The party that initiates the hold is called the holding party; the other party is called the held party.
- After a successful Hold, the holding party hears a Dial Tone (HELD_TONE, defined in gateway's Call Progress Tones file).
- Call retrieve can be performed only by the holding party while the call is held and active.
- The holding party performs the retrieve by pressing the hook-flash.
- After a successful retrieve, voice is connected again.
- The hold and retrieve functionalities are implemented by REINVITE messages. The IP address 0.0.0.0 as the connection IP address or the string 'a=inactive' in the received REINVITE SDP cause the gateway to enter Hold state and to play held tone (configured in the gateway) to the PBX/PSTN. If the string 'a=recvonly' is received in the SDP message, the gateway stops sending RTP packets, but continues to listen to the incoming RTP packets. Usually, the remote party plays, in this scenario, Music on Hold (MOH) and the Mediant 1000 forwards the MOH to the held party

9.1.1.2 Receiving Hold / Retrieve

- When an active call receives REINVITE message with either the IP address 0.0.0.0 or the 'inactive' string in SDP, the gateway stops sending RTP and plays a local Held Tone.
- When an active call receives REINVITE message with 'sendonly' string in SDP, the gateway stops sending RTP and listens to the remote party. In this mode, it is expected that on-hold music (or any other hold tone) is to be played (over IP) by the remote party.

9.1.2 Consultation / Alternate

- The consultation feature is relevant only for the holding party (applicable only to the Mediant 1000/FXS module).
- After holding a call (by pressing hook-flash), the holding party hears a dial tone and can now initiate a new call, which is called a consultation call.
- While hearing a dial tone, or when dialing to the new destination (before dialing is complete), the user can retrieve the held call by pressing hook-flash.
- The held call can't be retrieved while Ringback tone is heard.
- After the consultation call is connected, the user can switch between the held and active call by pressing hook-flash.

9.1.3 Call Transfer

There are two types of call transfers:

- Consultation Transfer (REFER and REPLACES)
- Blind Transfer (REFER)

The common way to perform a consultation transfer is as follows:

In the transfer scenario there are three parties:

Party A = transferring, Party B = transferred, Party C = transferred to.

- A Calls B.
- B answers.
- A presses the hook-flash and puts B on-hold (party B hears a hold tone)
- A dials C.
- After A completes dialing C, A can perform the transfer by onhook of the A phone.
- After the transfer is completed, B and C parties are engaged in a call.

The transfer can be initiated at any of the following stages of the call between A to C:

- Just after completing dialing C phone number - Transfer from setup.
- While hearing Ringback – Transfer from alert.
- While speaking to C – Transfer from active.

Blind transfer is performed after we have a call between A and B, and party A decides to transfer the call to C immediately without speaking with C.

The result of the transfer is a call between B and C (just like consultation transfer only skipping the consultation stage).

Note the following SIP issues:

- Transfer is initiated by sending REFER with REPLACES.
- The gateway can receive and act upon receiving REFER with or without REPLACES.
- The gateway can receive and act upon receiving INVITE with REPLACES, in which case the old call is replaced by the new one.
- The INVITE with REPLACES can be used to implement Directed Call Pickup.

9.1.4 Call Forward

Five forms of call forward are supported:

- **Immediate:** Any incoming call is forwarded immediately and unconditionally.
- **Busy:** Incoming call is forwarded if the endpoint is busy.
- **No Reply:** The incoming call is forwarded if it isn't answered for a specified time.

- **On Busy or No Reply:** Forward incoming calls when the port is busy or when calls are not answered after a specified time.
- **Do Not Disturb:** Immediately reject incoming calls. Upon receiving a call to Do Not Disturb call, the 603 Decline SIP response code is sent.

Three forms of forwarding parties are available:

- **Served party:** the party that is configured to forward the call – Mediant 1000/FXS.
- **Originating party:** the party that initiated the first call – Mediant 1000/FXS or FXO.
- **Diverted party:** the new destination of the forwarded call – Mediant 1000/FXS or FXO.

The served party (Mediant 1000/FXS) can be configured through the Web browser (refer to Section 5.5.6.4 on page 124) or via *ini* file to activate one of the call forward modes. These modes are configurable per gateway's endpoint.

Note the following SIP issues:

- Initiating forward – When forward is initiated, the gateway sends a 302 response with a contact that contains the phone number from the forward table and its corresponding IP address from the routing table (or, when Proxy is used, the proxy's IP address).
- Receiving forward – The gateway handles 3xx responses for redirecting calls with a new contact.

9.1.5 Call Waiting

The Call Waiting feature enables FXS modules to accept an additional (second) call on busy endpoints. If an incoming IP call is designated to a busy port, the called party hears call waiting tone (several configurable short beeps) and (for Bellcore and ETSI Caller IDs) can view the Caller ID string of the incoming call. The calling party hears a Call Waiting Ringback Tone. Called party can accept the new call, using hook-flash, and can toggle between the two calls.

To enable Call Waiting:

- Set 'EnableCallWaiting = 1'.
- Set 'EnableHold = 1'.
- Define the Call Waiting indication and Call Waiting Ringback tones in the Call Progress Tones file. You can define up to four Call Waiting indication tones (refer to the parameter 'FirstCallWaitingToneID' in Table 6-7 on page 230).
- To configure the Call Waiting indication tone cadence, modify the following parameters: 'NumberOfWaitingIndications', 'WaitingBeepDuration' and 'TimeBetweenWaitingIndications'.
- To configure a delay interval before a Call Waiting Indication is played to the currently busy port use the parameter 'TimeBeforeWaitingIndication'. This enables the caller to hang up before disturbing the called party with Call Waiting Indications. Applicable only to FXS modules.

Both the calling and the called sides are supported by FXS modules; the FXO modules support only the calling side.

To indicate Call Waiting, the gateway sends a 182 - call queued response.

The gateway identifies a Waiting Call when a 182 (call queued response) is received.

9.1.6 Message Waiting Indication

Support for Message Waiting Indication (MWI) according to IETF <draft-ietf-sipping-mwi-04.txt>, including SUBSCRIBE (to MWI server). Mediant 1000/FXS modules can accept an MWI NOTIFY message that indicates waiting messages or that the MWI is cleared. Users are informed of these messages by a stutter dial tone. The stutter and confirmation tones are defined in the CPT file (refer to Section 17.1 on page 391). If the MWI display is configured, the number of waiting messages is also displayed. If the MWI lamp is configured, the phone's lamp (on a phone that is equipped with an MWI lamp) is lit. The gateway can subscribe to the MWI server per port (usually used on FXS) or per gateway (used on FXO).

To configure MWI, set the following parameters:

- EnableMWI
- MWIServerIP
- MWIAnalogLamp
- MWIDisplay
- StutterToneDuration
- EnableMWISubscription
- MWIExpirationTime
- SubscribeRetryTime
- SubscriptionMode
- CallerIDType (determines the standard for detection of MWI signals)
- ETSIVMWITypeOneStandard
- BellcoreVMWITypeOneStandard

9.2 Configuring the DTMF Transport Types

You can control the way DTMF digits are transported over the IP network to the remote endpoint. The following five modes are supported:

1. Using INFO message according to the Nortel IETF draft:
   In this mode DTMF digits are carried to the remote side within INFO messages. To enable this mode set:
   - 'RxDTMFOption = 0' (**Protocol Management** menu > **Protocol Definition** submenu > **DTMF & Dialing** option > 'Declare RFC 2833 in SDP' = No))
   - 'TxDTMFOption = 1' (1st to 5th DTMF Option = INFO (Nortel))

   Note that in this mode DTMF digits are erased from the audio stream (DTMFTransportType is automatically set to 0 (DTMF Mute)).
2. Using INFO message according to Cisco's mode:
   In this mode DTMF digits are carried to the remote side within INFO messages. To enable this mode set:
   - 'RxDTMFOption = 0' (Declare RFC 2833 in SDP = No)
   - 'TxDTMFOption = 3' (1st to 5th DTMF Option = INFO (Cisco))

   Note that in this mode DTMF digits are erased from the audio stream (DTMFTransportType is automatically set to 0 (DTMF Mute)).

3. Using NOTIFY messages according to <draft-mahy-sipping-signaled-digits-01.txt>: In this mode DTMF digits are carried to the remote side using NOTIFY messages. To enable this mode set:

   - 'RxDTMFOption = 0' (Declare RFC 2833 in SDP = No)
   - 'TxDTMFOption = 2' (1st to 5th DTMF Option = NOTIFY)

   Note that in this mode DTMF digits are erased from the audio stream (DTMFTransportType is automatically set to 0 (DTMF Mute)).

4. Using RFC 2833 relay with Payload type negotiation: In this mode, DTMF digits are carried to the remote side as part of the RTP stream in accordance with RFC 2833 standard. To enable this mode set:

   - 'TxDTMFOption = 4' (1st to 5th DTMF Option = RFC 2833)
   - 'RxDTMFOption = 3' (Declare RFC 2833 in SDP = Yes)

   Note that to set the RFC 2833 payload type with a different value (other than its default, 96) configure the 'RFC2833PayloadType' (RFC 2833 Payload Type) parameter. The gateway negotiates the RFC 2833 payload type using local and remote SDP and sends packets using the PT from the received SDP. The gateway expects to receive RFC 2833 packets with the same PT as configured by the 'RFC2833PayloadType' parameter. If the remote side doesn't include 'telephony-event' in its SDP, the gateway sends DTMF digits in transparent mode (as part of the voice stream).

5. Sending DTMF digits (in RTP packets) as part of the audio stream (DTMF Relay is disabled): Note that this method is normally used with G.711 coders; with other LBR coders the quality of the DTMF digits is reduced. To ser this mode:

   - 'TxDTMFOption = 0' (1st to 5th DTMF Option = Disable)
   - 'RxDTMFOption = 0' (Declare RFC 2833 in SDP = No)
   - 'DTMFTransportType = 2' (DTMF Transport Type = Transparent DTMF)

Note 1: The gateway is always ready to receive DTMF packets over IP, in all possible transport modes: INFO messages, NOTIFY and RFC 2833 (in proper payload type) or as part of the audio stream.

Note 2: To exclude RFC 2833 Telephony event parameter from the gateway's SDP, set 'RxDTMFOption = 0' in the *ini* file.

The following parameters affect the way the Mediant 1000 SIP handles the DTMF digits:

- TxDTMFOption, RxDTMFOption and RFC2833PayloadType (described in Table 5-6).
- MGCPDTMFDetectionPoint, DTMFDigitLength and DTMFInterDigitInterval (Section 6.5.13 on page 255).
- DTMFVolume and DTMFTransportType (Table 5-28).

9.3 Fax & Modem Transport Modes

9.3.1 Fax/Modem Settings

Users may choose to use one of the following transport methods for fax and for each modem type (V.22/V.23/Bell/V.32/V.34):

- **Fax relay:** demodulation / modulation
- **Bypass:** using a high bit rate coder to pass the signal
- **Transparent:** passing the signal in the current voice coder

When the fax relay mode is enabled, distinction between fax and modem is not immediately possible at the beginning of a session. The channel is therefore in 'Answer Tone' mode until a distinction is determined. The packets being sent to the network at this stage are T.38-complaint fax relay packets.

9.3.2 Configuring Fax Relay Mode

When FaxTransportMode = 1 (relay mode), then on detection of fax the channel automatically switches from the current voice coder to answer tone mode, and then to T.38-compliant fax relay mode.

When fax transmission has ended, the reverse switching from fax relay to voice is performed. This mode switching automatically occurs at both the local and remote endpoints.

Users can limit the fax rate using the FaxRelayMaxRate parameter and can enable/disable ECM fax mode using the FaxRelayECMEnable parameter.

When using T.38 mode, the user can define a redundancy feature to improve fax transmission over congested IP network. This feature is activated by 'FaxRelayRedundancyDepth' and 'FaxRelayEnhancedRedundancyDepth' parameters. Although this is a proprietary redundancy scheme, it should not create problems when working with other T.38 decoders.

Note: T.38 mode currently supports only the T.38 UDP syntax.

9.3.3 Configuring Fax/Modem Bypass Mode

When VxxTransportType= 2 (FaxModemBypass, Vxx can be one of the following: V32/V22/Bell/V34/Fax), then on detection of fax/modem, the channel automatically switches from the current voice coder to a high bit-rate coder (G.711 or G.726), as defined by the user, with the FaxModemBypassCoderType configuration parameter.

During the bypass period, the coder uses the packing factor (by which a number of basic coder frames are combined together in the outgoing WAN packet) set by the user in the FaxModemBypassM configuration parameter. The network packets generated and received during the bypass period are regular voice RTP packets (per the selected bypass coder) but with a different RTP Payload type.

When fax/modem transmission ends, the reverse switching, from bypass coder to regular voice coder, is carried out.

9.3.4 Supporting V.34 Faxes

V.34 faxes don’t comply with the T.38 relay standard. We therefore provide the optional modes described under Sections 9.3.4.1 and 9.3.4.2:

Note that the CNG detector is disabled (CNGDetectorMode=0) in all the following examples.

9.3.4.1 Using Bypass Mechanism for V.34 Fax Transmission

In this proprietary scenario, the media gateway uses a high bit-rate coder to transmit V.34 faxes, enabling the full utilization of its speed.

Refer to the following configurations:

```
FaxTransportMode = 2 (Bypass)
V34ModemTransportType = 2 (Modem bypass)
V32ModemTransportType = 2
V23ModemTransportType = 2
V22ModemTransportType = 2
```

In this configuration, both T.30 and V.34 faxes work in Bypass mode.

Or

```
FaxTransportMode = 1 (Relay)
V34ModemTransportType = 2 (Modem bypass)
V32ModemTransportType = 2
V23ModemTransportType = 2
V22ModemTransportType = 2
```

In this configuration, T.30 fax uses T.38 Relay mode while V.34 fax uses Bypass mode.

9.3.4.2 Using Relay mode for both T.30 and V.34 faxes

In this scenario, V.34 fax machines are compelled to use their backward compatibility with T.30 faxes; as a T.30 machine, the V.34 fax can use T.38 Relay mode.

Refer to the following configuration:

```
FaxTransportMode = 1 (Relay)
V34ModemTransportType = 0 (Transparent)
V32ModemTransportType = 0
V23ModemTransportType = 0
V22ModemTransportType = 0
```

Both T.30 and V.34 faxes use T.38 Relay mode. This configuration forces the V.34 fax machine to operate in the slower T.30 mode.

9.3.5 Supporting V.152 Implementation

The gateway supports the ITU-T recommendation V.152 (Procedures for Supporting Voice-Band Data over IP Networks). Voice-band data (VBD) is the transport of modem, facsimile, and text telephony signals over a voice channel of a packet network with a codec appropriate for such signals.

For V.152 capability, the gateway supports T.38 as well as VBD codecs (i.e., G.711 A-law and G.711 µ-law). The selection of capabilities is performed using the coders table.

When in VBD mode for V.152 implementation, support is negotiated between the gateway and the remote endpoint at the establishment of the call. During this time, initial exchange of call capabilities is exchanged in the outgoing SDP. These capabilities include whether VBD is supported and associated RTP payload types ('gpmd' SDP attribute), supported codecs, and packetization periods for all codec payload types ('ptime' SDP attribute). After this initial negotiation, no Re-INVITE messages are necessary as both endpoints are synchronized in terms of the other side's capabilities. If negotiation fails (i.e., no match was achieved for any of the transport capabilities), fallback to existing logic occurs (according to the parameter IsFaxUsed).

Below is an example of media descriptions of an SDP indicating support for V.152.

```
v=0
o=-  0 0 IN IPV4 <IPAdressA>
s=-
t=0 0
p=+1
c=IN IP4  <IPAddressA
m=audio <udpPort A> RTP/AVP 18 0
a=ptime:10
a=rtpmap:96 PCMU/8000
a=gpmd: 96 vbd=yes
```

In the example above, V.152 implementation is supported (using the dynamic payload type 96 and G.711 mu-law as the VBD codec) as well as the voice codecs G.711 mu-law and G.729.

Instead of using VBD transport mode, the V.152 implementation can use alternative relay fax transport methods (e.g., fax relay over IP using T.38). The preferred V.152 transport method is indicated by the SDP ‘pmft’ attribute. Omission of this attribute in the SDP content means that VBD mode is the preferred transport mechanism for voice-band data.

9.4 FXO Operating Modes

This section provides a description of the FXO operating modes and gateway configurations for Tel-to-IP and IP-to-Tel calls.

9.4.1 IP-to-Telephone Calls

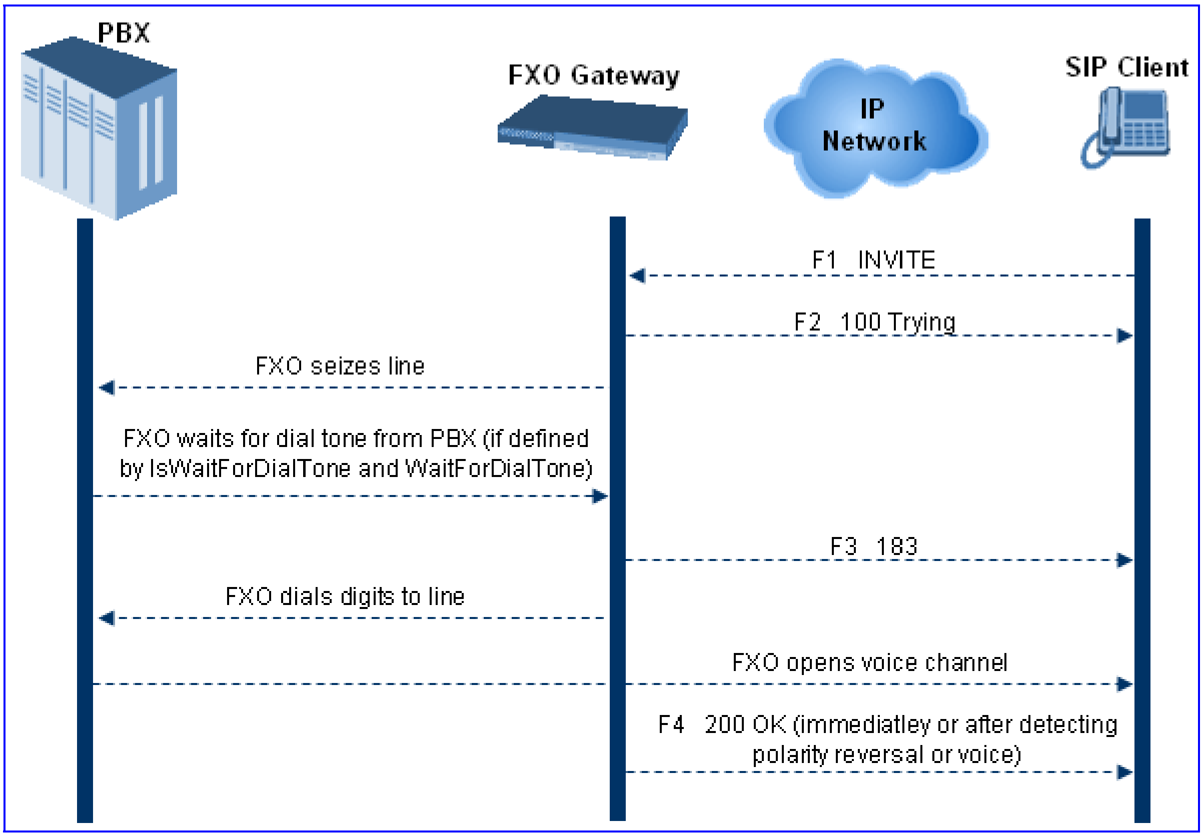
The FXO gateway provides the following operating modes for IP-to-Tel calls:

- One-stage dialing
  - Waiting for dial tone
  - Time to wait before dialing
  - Answer supervision
- Two-stage dialing
- Dialing time
  - Disconnect supervision
  - DID wink

9.4.1.1 One-Stage Dialing

One-stage dialing is when the FXO gateway receives an IP-to-Tel call, off-hooks the PBX line connected to the telephone, and then immediately dials the destination telephone number. In other words, the IP caller doesn't dial the PSTN number upon hearing a dial tone.

Figure 9-1: Call Flow for One-Stage Dialing


PBX
FXO Gateway
IP Network
SIP Client
F1 INVITE
F2 100 Trying
FXO seizes line
FXO waits for dial tone from PBX (if defined by IsWaitForDialTone and WaitForDialTone)
F3 183
FXO dials digits to line
FXO opens voice channel
F4 200 OK (immediatley or after detecting polarity reversal or voice)

One -stage dialing incorporates the following FXO functionality:

- **Waiting for Dial Tone**

  The Waiting for Dial Tone feature enables the gateway to dial the digits to the Tel side only after detecting a dial tone from the PBX line. The *ini* file parameter IsWaitForDialTone is used to configure this operation.

- **Time to Wait Before Dialing**

  The Time to Wait Before Waiting feature defines the time (in msec) between seizing the FXO line and starting to dial the digits. The *ini* file parameter WaitForDialTime is used to configure this operation.

Note: The *ini* file parameter IsWaitForDialTone must be disabled for this mode.

- **Answer Supervision**

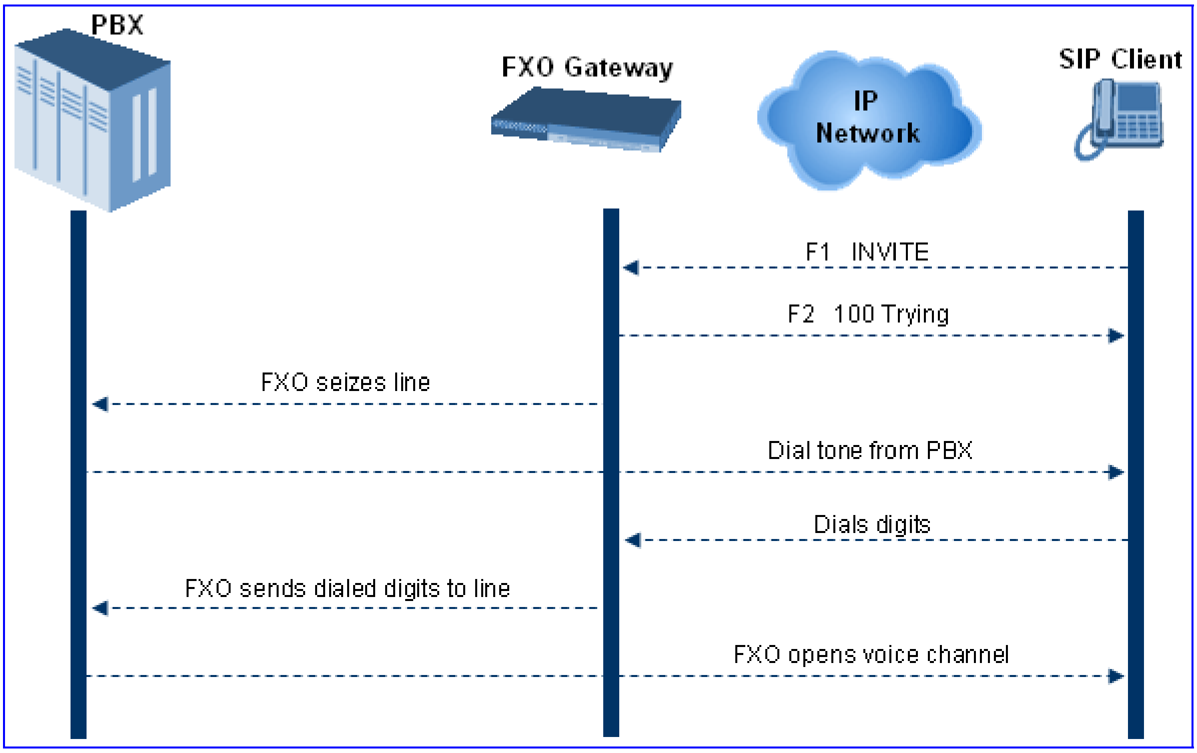
  The Answer Supervision feature enables the FXO gateway to determine when a call is connected, by using one of the following methods:

  - Polarity Reversal: the gateway sends a 200 OK in response to an INVITE only when it detects a polarity reversal.
  - Voice Detection: the gateway sends a 200 OK in response to an INVITE only when it detects the start of speech (or ringback tone) from the Tel side. (Note that the IPM detectors must be enabled).

9.4.1.2 Two-Stage Dialing

Two-stage dialing is when the IP caller is required to dial twice. The caller initially dials to the FXO gateway, and only after receiving a dial tone from the PBX (via the FXO gateway), dials the destination telephone number.

Figure 9-2: Call Flow for Two-Stage Dialing



Two-stage dialing implements the Dialing Time feature. Dialing Time allows you to define the time that each digit can be separately dialed. By default, the overall dialing time per digit is 200 msec. The longer the telephone number, the greater the dialing time will be.

The relevant parameters for configuring Dialing Time include the following:

- DTMFDigitLength (100 msec): time for generating DTMF tones to the PSTN (PBX) side
- DTMFInterDigitInterval (100 msec): time between generated DTMF digits to PSTN (PBX) side

9.4.1.3 Call Termination (Disconnect Supervision) on Mediant 1000/FXO

The FXO Disconnect Supervision enables the gateway's FXO ports to monitor call-progress tones from a PBX or from the PSTN. This allows the FXO to determine when the call has terminated on the PBX side, and thereby, prevents analog trunks (i.e., lines to the PBX) from getting “stuck” when the called phone hangs up.

The PBX doesn't disconnect the call, but instead signals to the gateway that the call is disconnected using one of the following methods:

- **Detection of polarity reversal / current disconnect**

  This is the recommended method. The call is immediately disconnected after polarity reversal or current disconnect is detected on the Tel side (assuming the PBX / CO produces this signal).

  Relevant parameters: EnableReversalPolarity, EnableCurrentDisconnect, CurrentDisconnectDuration, CurrentDisconnectDefaultThreshold, and TimeToSampleAnalogLineVoltage.

- **Detection of Reorder / Busy / Dial tones**

  The call is immediately disconnected after Reorder / Busy / Dial tone is detected on the Tel side (assuming the PBX / CO produces this tone). This method requires the correct tone frequencies and cadence to be defined in the Call Progress Tones file. If these frequencies are not known, define them in the CPT file (the tone produced by the PBX / CO must be recorded and its frequencies analyzed -- refer to Section G.2.7 on page 441). This method is slightly less reliable than the previous one. You can use the CPTWizard (described in Section G.1.4 on page 433) to analyze Call Progress Tones generated by any PBX or telephone network.

  Relevant parameters: DisconnectOnBusyTone and DisconnectOnDialTone.

- **Detection of silence**

  The call is disconnected after silence is detected on both call directions for a specific (configurable) amount of time. The call isn’t disconnected immediately; therefore, this method should only be used as a backup option.

  Relevant parameters: EnableSilenceDisconnect and FarEndDisconnectSilencePeriod (with DSP templates number 2 or 3).

- **A special DTMF code**

  A digit pattern that when received from the Tel side, indicates the gateway to disconnect the call.

  Relevant *ini* file parameter: TelDisconnectCode.

- **Interruption of RTP stream:**

  Relevant parameters: BrokenConnectionEventTimeout and DisconnectOnBrokenConnection.

Note: This method operates correctly only if silence suppression is not used.

- **Protocol-based termination of the call from the IP side**

Note: The implemented disconnect method must be supported by the CO or PBX.

9.4.1.4 DID Wink

The gateway's FXO ports support Direct Inward Dialing (DID). DID is a service offered by telephone companies that enables callers to dial directly to an extension on a PBX without the assistance of an operator or automated call attendant. This service makes use of DID trunks, which forward only the last three to five digits of a phone number to the PBX. If, for example, a company has a PBX with extensions 555-1000 to 555-1999, and a caller dials 555-1234, the local central office (CO) would forward, for example, only 234 to the PBX. The PBX would then ring extension 234.

DID wink enables the originating end to seize the line by going off-hook. It waits for acknowledgement from the other end before sending digits. This serves as an integrity check that identifies a malfunctioning trunk and allows the network to send a re-order tone to the calling party.

The "start dial" signal is a wink from the PBX to the FXO gateway. The FXO then sends the last four to five DTMF digits of the called number. The PBX uses these digits to complete the routing directly to an internal station (telephone or equivalent)

- DID Wink can be used for connection to EIA/TIA-464B DID Loop Start lines
- Both FXO (detection) and FXS (generation) are supported

9.4.2 Telephone-to-IP Calls

The FXO gateway provides the following FXO operating modes for Tel-to-IP calls:

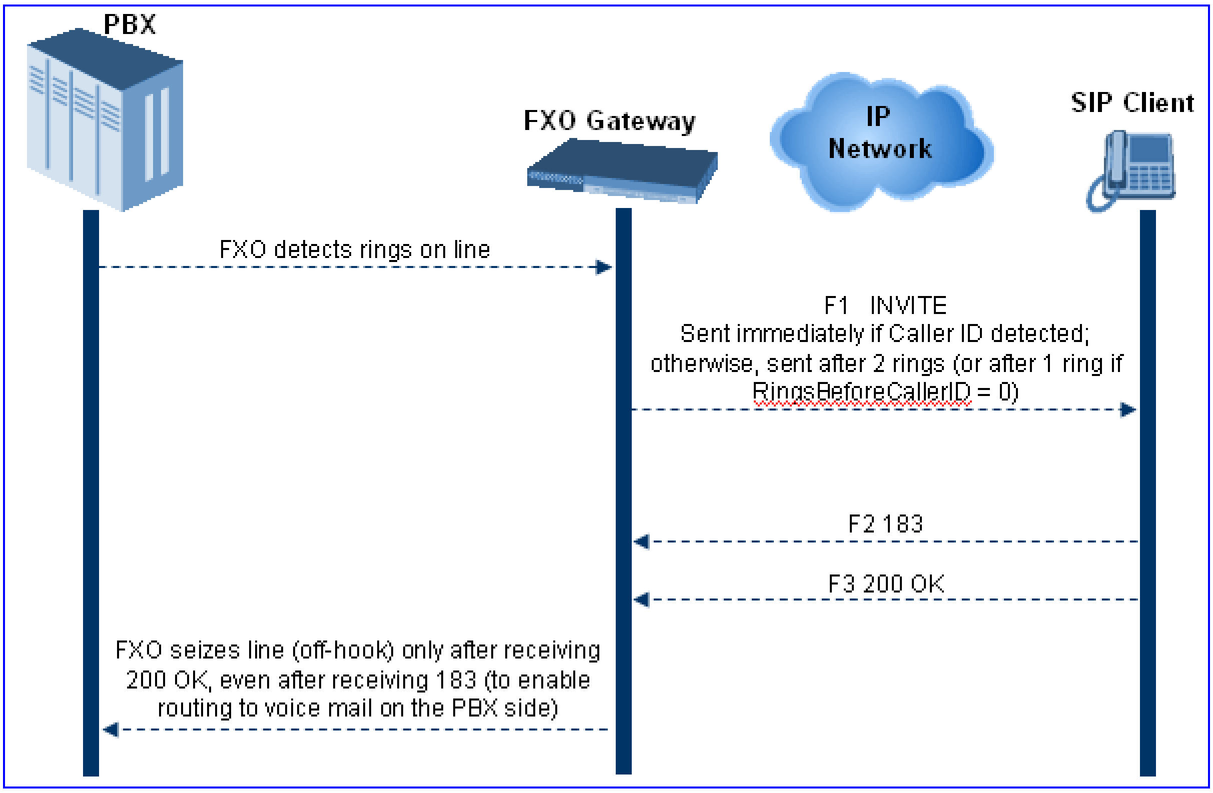
- Automatic Dialing
- Collecting Digits Mode
- Ring Detection Timeout
- FXO Supplementary Services
  - Hold/Transfer Toward the Tel side
  - Hold/Transfer Toward the IP side
  - Blind Transfer to the Tel side

9.4.2.1 Automatic Dialing

Automatic dialing is defined using the *ini* file parameter TargetOfChannelX (where 'X' is the channel number) or the embedded Web server's 'Automatic Dialing' screen (refer to Section 5.5.6.2 on page 122).

The SIP call flow diagram below illustrates Automatic Dialing.

Figure 9-3: Call Flow for Automatic Dialing

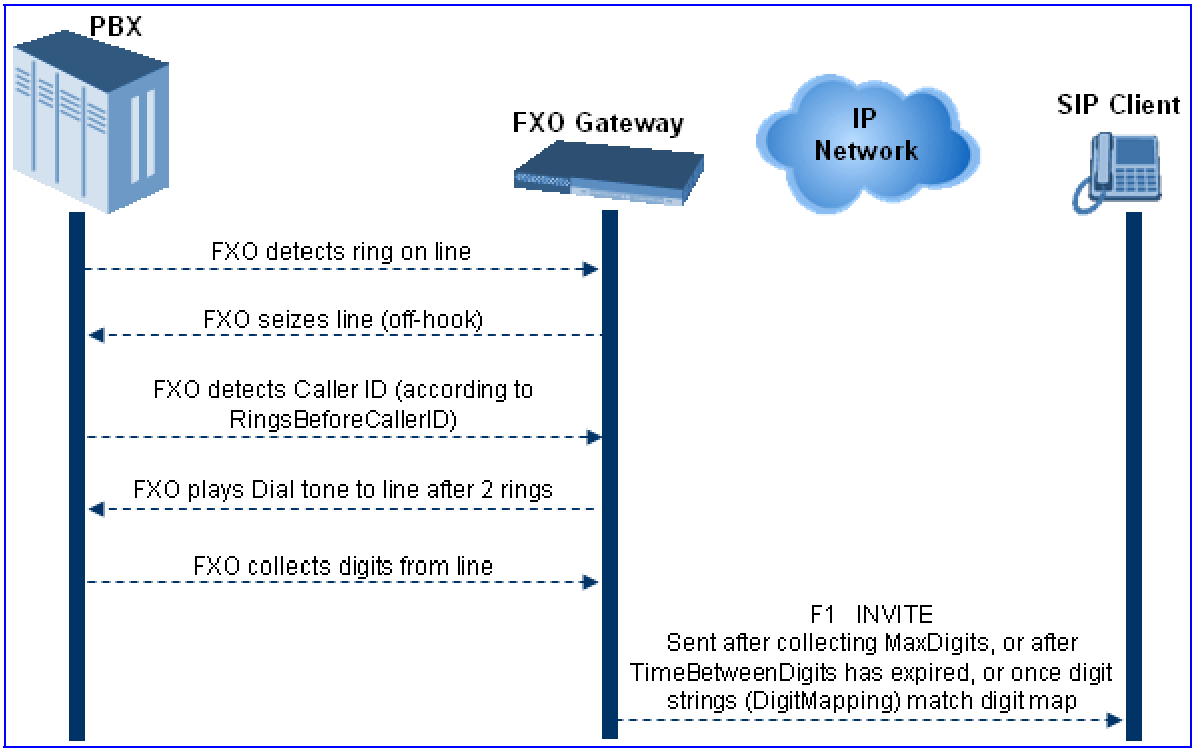

PBX
FXO Gateway
IP Network
SIP Client
FXO detects rings on line
F1 INVITE
Sent immediately if Caller ID detected; otherwise, sent after 2 rings (or after 1 ring if RingsBeforeCallerID = 0)
F2 183
F3 200 OK
FXO seizes line (off-hook) only after receiving 200 OK, even after receiving 183 (to enable routing to voice mail on the PBX side)

9.4.2.2 Collecting Digits Mode

When automatic dialing is not defined, the gateway collects the digits.

The SIP call flow diagram below illustrates the Collecting Digits Mode.

Figure 9-4: Call Flow for Collecting Digits Mode


PBX
FXO Gateway
IP Network
SIP Client
FXO detects ring on line
FXO seizes line (off-hook)
FXO detects Caller ID (according to RingsBeforeCallerID)
FXO plays Dial tone to line after 2 rings
FXO collects digits from line
F1 INVITE
Sent after collecting MaxDigits, or after TimeBetweenDigits has expired, or once digit strings (DigitMapping) match digit map

9.4.2.3 Ring Detection Timeout

The *ini* file parameters IsWaitForDialTone and WaitForDialTone apply to Ring Detection Timeout. The operation of Ring Detection Timeout depends on the following:

- No automatic dialing and Caller ID is enabled: if the second ring signal doesn't arrive for Ring Detection Timeout, the gateway doesn't initiate a call to the IP.
- Automatic dialing is enabled: if the remote party doesn't answer the call, and the ringing signal stops for Ring Detection Timeout, the FXO releases the IP call.

Ring Detection Timeout supports full ring cycle of ring on and ring off (from ring start to ring start).

9.4.2.4 FXO Supplementary Services

- **Hold / Transfer toward the Tel side**

  The *ini* file parameter LineTransferMode must be set to 0 (default).

  If the FXO receives a hook-flash from the IP side (using out-of-band or RFC 2833), the gateway sends the hook-flash to the Tel side by one of the following:

  - Performing a hook flash (i.e., on-hook and off-hook)
  - Sending a hook-flash code (defined by the *ini* file parameter HookFlashCode)

  The PBX may generate a dial tone that is sent to the IP, and the IP side may dial digits of a new destination.

- **Blind Transfer to the Tel side**

  A blind transfer is one in which the transferring phone connects the caller to a destination line before ringback begins. The *ini* file parameter LineTransferMode must be set to 1.

  The blind transfer call process is as follows:

  - FXO receives a REFER request from the IP side
  - FXO sends a hook-flash to the PBX, dials the digits (that are received in the Refer-To header), and then drops the line (on-hook). Note that the time between flash to dial is according to the WaitForDialTime parameter.
  - PBX performs the transfer internally

- **Hold / Transfer toward the IP side**

  The FXO gateway doesn't initiate hold / transfer as a response to input from the Tel side. If the FXO receives a REFER request (with or without replaces), it generates a new INVITE according to the Refer-To header.

9.5 Event Notification using X-Detect Header

The gateway supports the sending of notifications to a remote party notifying the occurrence (or detection) of certain events on the media stream. Event detection and notifications is performed using the X-Detect SIP message header, and only when establishing a SIP dialog.

For supporting some events, certain gateway configurations need to be performed. The table below lists the support event types (and subtypes) and the corresponding gateway configurations, if required:

Table 9-1: Supported X-Detect Event Types

Events		Required Gateway Configuration
Type	Subtype	

AMD	voice
	automatic
	silence
	unknown
		EnableDSPIPMDetectors = 1
		AMDTimeout = 2000 (msec)

CPT	SIT	SITDetectorEnable = 1
		UserDefinedToneDetectorEnable = 1
		Note: Differentiation of SIT is not supported in 5.0.

FAX	CED	(IsFaxUsed ≠ 0) or (IsFaxUsed = 0 and FaxTransportMode ≠ 0)
	modem	VxxModemTransportType = 3
PTT	voice-start voice-end	EnableDSPIPMDetectors = 1

The X-Detect event notification process is as follows:

1. For IP-to-Tel or Tel-to-IP calls, the gateway receives a SIP request message (using the X-Detect header) that the remote party wishes to detect events on the media stream. For incoming (IP-to-Tel) calls, the request must be indicated in the initial INVITE and responded to either in the 183 response (for early dialogs) or in the 200 OK response (for confirmed dialogs). For outgoing calls (Tel-to-IP), the request may be received in the 183 (for early dialogs) and responded to in the PRACK, or received in the 200 OK (for confirmed dialogs) and responded to in the ACK.
2. Once the gateway receives such a request, it sends a SIP response message (using the X-Detect header) to the remote party, listing all supported events that can be detected. The absence of the X-Detect header indicates that no detections are available.
3. Each time the gateway detects a supported event, the event is notified to the remote party, by sending an INFO message with the following message body:
    - Content-Type: application/X-DETECT
    - Type = [AMD | CPT | FAX | PTT…]
    - Subtype = xxx (according to the defined subtypes of each type)

Below is an example of SIP messages implementing the X-Detect header:

```
INVITE sip:101@10.33.2.53;user=phone SIP/2.0
Via: SIP/2.0/UDP 10.33.2.53;branch=z9hG4bKac5906
Max-Forwards: 70
From: "anonymous" <sip:anonymous@anonymous.invalid>;tag=1c25298
To: <sip:101@10.33.2.53;user=phone>
Call-ID: 11923@10.33.2.53
CSeq: 1 INVITE
Contact: <sip:100@10.33.2.53>
X- Detect: Request=CPT,FAX

SIP/2.0 200 OK
Via: SIP/2.0/UDP 10.33.2.53;branch=z9hG4bKac5906
From: "anonymous" <sip:anonymous@anonymous.invalid>;tag=1c25298
To: <sip:101@10.33.2.53;user=phone>;tag=1c19282
Call-ID: 11923@10.33.2.53
CSeq: 1 INVITE
Contact: <sip:101@10.33.2.53>
X- Detect: Response=CPT,FAX

INFO sip:101@10.33.2.53;user=phone SIP/2.0
Via: SIP/2.0/UDP 10.33.2.53;branch=z9hG4bKac5906
Max-Forwards: 70
From: "anonymous" <sip:anonymous@anonymous.invalid>;tag=1c25298
To: <sip:101@10.33.2.53;user=phone>
Call-ID: 11923@10.33.2.53
CSeq: 1 INVITE
Contact: <sip:100@10.33.2.53>
X- Detect: Response=CPT,FAX
Content-Type: Application/X-Detect
Content-Length: xxx
Type = CPT
Subtype = SIT
```

9.6 ThroughPacket™

The gateway supports a proprietary method to aggregate RTP streams from several channels to reduce the bandwidth overhead caused by the attached Ethernet, IP, UDP and RTP headers, and to reduce the packet / data transmission rate. This option reduces the load on network routers and can typically save 50% (e.g., for G.723) on IP bandwidth.

ThroughPacket™ is accomplished by aggregating payloads from several channels that are sent to the same destination IP address into a single IP packet.

ThroughPacket™ can be applied to the entire gateway or, using IP Profile, to specific IP destinations (refer to Section 5.5.5.3 on page 119). Note that ThroughPacket™ must be enabled on both gateways.

To enable ThroughPacket™ set the parameter 'RemoteBaseUDPPort' to a nonzero value. Note that the value of 'RemoteBaseUDPPort' on the local gateway must equal the value of 'BaseUDPPort' of the remote gateway. The gateway uses these parameters to identify and distribute the payloads from the received multiplexed IP packet to the relevant channels.

In ThroughPacket™ mode, the gateway uses a single UDP port for all incoming multiplexed packets and a different port for outgoing packets. These ports are configured using the parameters 'L1L1ComplexTxUDPPort' and 'L1L1ComplexRxUDPPort'.

When ThroughPacket™ is used, Call statistics aren't available (since there is no RTCP flow).

9.7 Dynamic Jitter Buffer Operation

Voice frames are transmitted at a fixed rate. If the frames arrive at the other end at the same rate, voice quality is perceived as good. In many cases, however, some frames can arrive slightly faster or slower than the other frames. This is called jitter (delay variation), and degrades the perceived voice quality. To minimize this problem, the gateway uses a jitter buffer. The jitter buffer collects voice packets, stores them and sends them to the voice processor in evenly spaced intervals.

The Mediant 1000 uses a dynamic jitter buffer that can be configured using two parameters:

- Minimum delay, 'DJBufMinDelay' (0 msec to 150 msec):
  Defines the starting jitter capacity of the buffer. For example, at 0 msec, there is no buffering at the start. At the default level of 10 msec, the gateway always buffers incoming packets by at least 10 msec worth of voice frames.

- Optimization Factor, 'DJBufOptFactor' (0 to 12, 13):
  Defines how the jitter buffer tracks to changing network conditions. When set at its maximum value of 12, the dynamic buffer aggressively tracks changes in delay (based on packet loss statistics) to increase the size of the buffer and doesn't decays back down. This results in the best packet error performance, but at the cost of extra delay. At the minimum value of 0, the buffer tracks delays only to compensate for clock drift and quickly decays back to the minimum level. This optimizes the delay performance but at the expense of a higher error rate.

The default settings of 10 msec Minimum delay and 10 Optimization Factor should provide a good compromise between delay and error rate. The jitter buffer 'holds' incoming packets for 10 msec before making them available for decoding into voice. The coder polls frames from the buffer at regular intervals in order to produce continuous speech. As long as delays in the network do not change (jitter) by more than 10 msec from one packet to the next, there is always a sample in the buffer for the coder to use. If there is more than 10 msec of delay at any time during the call, the packet arrives too late. The coder tries to access a frame and is not able to find one. The coder must produce a voice sample even if a frame is not available. It therefore compensates for the missing packet by adding a Bad-Frame-Interpolation (BFI) packet. This loss is then flagged as the buffer being too small. The dynamic algorithm then causes the size of the buffer to increase for the next voice session. The size of the buffer may decrease again if the gateway notices that the buffer is not filling up as much as expected. At no time does the buffer decrease to less than the minimum size configured by the Minimum delay parameter.

Special Optimization Factor Value: 13

One of the purposes of the Jitter Buffer mechanism is to compensate for clock drift. If the two sides of the VoIP call are not synchronized to the same clock source, one RTP source generates packets at a lower rate, causing under-runs at the remote Jitter Buffer. In normal operation (optimization factor 0 to 12), the Jitter Buffer mechanism detects and compensates for the clock drift by occasionally dropping a voice packet or by adding a BFI packet.

Fax and modem devices are sensitive to small packet losses or to added BFI packets. Therefore to achieve better performance during modem and fax calls, the Optimization Factor should be set to 13. In this special mode the clock drift correction is performed less frequently - only when the Jitter Buffer is completely empty or completely full. When such condition occurs, the correction is performed by dropping several voice packets simultaneously or by adding several BFI packets simultaneously, so that the Jitter Buffer returns to its normal condition.

9.8 Configuring the Gateway's Alternative Routing (Based on Connectivity and QoS)

The Alternative Routing feature enables reliable routing of Tel to IP calls when a Proxy isn't used. The Mediant 1000 gateway periodically checks the availability of connectivity and suitable Quality of Service (QoS) before routing. If the expected quality cannot be achieved, an alternative IP route for the prefix (phone number) is selected.

Note: If the alternative routing destination is the gateway itself, the call can be configured to be routed back to one of the gateway's trunk groups and thus, back into the PSTN (PSTN Fallback).

9.8.1 Alternative Routing Mechanism

When a Tel→IP call is routed through the Mediant 1000 gateway, the call's destination number is compared to the list of prefixes defined in the Tel to IP Routing table (described in Section 5.5.4.2 on page 107). The Tel to IP Routing table is scanned for the destination number's prefix starting at the top of the table. When an appropriate entry (destination number matches one of the prefixes) is found; the prefix's corresponding destination IP address is checked. If the destination IP address is disallowed, an alternative route is searched for in the following table entries.

Destination IP address is disallowed if no ping to the destination is available (ping is continuously initiated every seven seconds), when an inappropriate level of QoS was detected, or when DNS host name is not resolved. The QoS level is calculated according to delay or packet loss of previously ended calls. If no call statistics are received for two minutes, the QoS information is reset.

The Mediant 1000 gateway matches the rules starting at the top of the table. For this reason, enter the main IP route above any alternative route.

9.8.2 Determining the Availability of Destination IP Addresses

To determine the availability of each destination IP address (or host name) in the routing table, one (or all) of the following (configurable) methods are applied:

- **Connectivity:** The destination IP address is queried periodically (currently only by ping).
- **QoS:** The QoS of an IP connection is determined according to RTCP statistics of previous calls. Network delay (in msec) and network packet loss (in percentage) are separately quantified and compared to a certain (configurable) threshold. If the calculated amounts (of delay or packet loss) exceed these thresholds the IP connection is disallowed.
- **DNS resolution:** When host name is used (instead of IP address) for the destination route, it is resolved to an IP address by a DNS server. Connectivity and QoS are then applied to the resolved IP address.

9.8.3 PSTN Fallback as a Special Case of Alternative Routing

The purpose of the PSTN Fallback feature is to enable the Mediant 1000 gateway to redirect PSTN originated calls back to the legacy PSTN network if a destination IP route is found unsuitable (disallowed) for voice traffic at a specific time.

To enable PSTN fallback, assign the IP address of the gateway itself as an alternative route to the desired prefixes. Note that calls (now referred to as IP to Tel calls) can be re-routed to a specific trunk group using the Routing parameters.

9.8.4 Relevant Parameters

The following parameters (described in Table 5-14) are used to configure the Alternative Routing mechanism:

- AltRoutingTel2IPEnable
- AltRoutingTel2IPMode
- IPConnQoSMaxAllowedPL
- IPConnQoSMaxAllowedDelay

9.9 Mapping PSTN Release Cause to SIP Response

The Mediant 1000 FXO module is used to interoperate between the SIP network and the PSTN/PBX. This interoperability includes the mapping of PSTN/PBX Call Progress Tones to SIP 4xx or 5xx responses for IP→Tel calls. The converse is also true: For Tel→IP calls, the SIP 4xx or 5xx responses are mapped to tones played to the PSTN/PBX.

When establishing an IP→Tel call, the following rules are applied:

If the remote party (PSTN/PBX) is busy and the FXO module detects a Busy tone, it sends 486 busy to IP. If it detects a Reorder tone, it sends 404 not found (no route to destination) to IP. In both cases the call is released. Note that if ‘DisconnectOnBusyTone = 0’ the FXO module ignores the detection of Busy/Reorder tones and doesn’t release the call.

For all other Mediant 1000 FXS/FXO release types (caused when there are no free channels in the specific trunk group, or when an appropriate rule for routing the call to a trunk group doesn’t exist, or if the phone number isn’t found), the Mediant 1000 sends SIP response (to IP) according to the parameter ‘DefaultReleaseCause’. This parameter defines Q.931 release causes. Its default value is ‘3’, which is mapped to SIP 404 response. By changing its value to ‘34’ SIP 503 response is sent. Other causes can be used as well.

9.10 Call Detail Report

The Call Detail Report (CDR) contains vital statistic information on calls made by the gateway. CDRs are generated at the end and (optionally) at the beginning of each call (determined by the parameter ‘CDRReportLevel’) and sent to a Syslog server. The destination IP address for CDR logs is determined by the parameter ‘CDRSyslogServerIP’.

Table 9-2 lists the CDR fields that are supported.

Table 9-2: Supported CDR Fields

Field Name	Description
Cid	Port Number
CallId	H.323/SIP Call Identifier
Trunk	Physical Trunk Number (digital only)
BChan	Selected B-Channel (digital only)
ConId	H.323/SIP Conference ID
TG	Trunk Group Number
EPTyp	Endpoint Type
Orig	Call Originator (IP, Tel)
SourceIp	Source IP Address
DestIp	Destination IP Address
TON	Source Phone Number Type
NPI	Source Phone Number Plan
SrcPhoneNum	Source Phone Number
SrcNumBeforeMap	Source Number Before Manipulation
TON	Destination Phone Number Type
NPI	Destination Phone Number Plan
DstPhoneNum	Destination Phone Number
DstNumBeforeMap	Destination Number Before Manipulation
Durat	Call Duration
Coder	Selected Coder
Intrv	Packet Interval
RtpIp	RTP IP Address
Port	Remote RTP Port
TrmSd	Initiator of Call Release (IP, Tel, Unknown)
TrmReason	Termination Reason
Fax	Fax Transaction during the Call
InPackets	Number of Incoming Packets
OutPackets	Number of Outgoing Packets
PackLoss	Number of Incoming Lost Packets
UniqueId	unique RTP ID
SetupTime	Call Setup Time
ConnectTime	Call Connect Time
ReleaseTime	Call Release Time
RTPdelay	RTP Delay
RTPjitter	RTP Jitter
RTPssrc	Local RTP SSRC
RemoteRTPssrc	Remote RTP SSRC
RedirectReason	Redirect Reason
TON	Redirection Phone Number Type
NPI	Redirection Phone Number Plan
RedirectPhonNum	Redirection Phone Number

9.11 Supported RADIUS Attributes

Use Table 9-3 below for explanations on the RADIUS attributes contained in the communication packets transmitted between the Mediant 1000 and a RADIUS Server.

Table 9-3: Supported RADIUS Attributes (continues on pages 288 to 289)

Attribute Number	Attribute Name	VSA No.	Purpose	Value Format	Sample	AAA[2]
Request Attributes						
1	User-Name		Account number or calling party number or blank	String up to 15 digits long	5421385747	Start Acc Stop Acc
4	NAS-IP-Address		IP address of the requesting Mediant 1000	Numeric	192.168.14.43	Start Acc Stop Acc
6	Service-Type		Type of service requested	Numeric	1: login	Start Acc Stop Acc
26	h323-incoming-conf-id	1	H.323/SIP call identifier	Up to 32 octets		Start Acc Stop Acc
26	h323-remote-address	23	IP address of the remote gateway	Numeric		Stop Acc
26	h323-conf-id	24	H.323/SIP call identifier	Up to 32 octets		Start Acc Stop Acc
26	h323-setup-time	25	Setup time in NTP format 1	String		Start Acc Stop Acc
26	h323-call-origin	26	The call's originator: Answering (IP) or Originator (PSTN)	String	Answer, Originate etc	Start Acc Stop Acc
26	h323-call-type	27	Protocol type or family used on this leg of the call	String	VoIP	Start Acc Stop Acc
26	h323-connect-time	28	Connect time in NTP format	String		Stop Acc
26	h323-disconnect-time	29	Disconnect time in NTP format	String		Stop Acc
26	h323-disconnect-cause	30	Q.931 disconnect cause code	Numeric		Stop Acc
26	h323-gw-id	33	Name of the gateway	String	SIPIDString	Start Acc Stop Acc
30	Called-Station-Id			String	8004567145	Start Acc
			Destination phone number	String	2427456425	Stop Acc
31	Calling-Station-Id		Calling Party Number (ANI)	String	5135672127	Start Acc Stop Acc
40	Acct-Status-Type		Account Request Type (start or stop) **Note:** 'start' isn't supported on the Calling Card application.	Numeric	1: start, 2: stop	Start Acc Stop Acc

[2] The values in column 'AAA' are as follows:
'Start Acc' - Start Accounting
'Stop Acc' - Stop Accounting

Table 9-3: Supported RADIUS Attributes (continues on pages 288 to 289)

Attribute Number	Attribute Name	VSA No.	Purpose	Value Format	Sample	AAA[2]
41	Acct-Delay-Time		No. of seconds tried in sending a particular record	Numeric	5	Start Acc Stop Acc
42	Acct-Input-Octets		Number of octets received for that call duration	Numeric		Stop Acc
43	Acct-Output-Octets		Number of octets sent for that call duration	Numeric		Stop Acc
44	Acct-Session-Id		A unique accounting identifier - match start & stop	String	34832	Start Acc Stop Acc
46	Acct-Session-Time		For how many seconds the user received the service	Numeric		Stop Acc
47	Acct-Input-Packets		Number of packets received during the call	Numeric		Stop Acc
48	Acct-Output-Packets		Number of packets sent during the call	Numeric		Stop Acc
61	NAS-Port-Type		Mediant 1000 physical port type on which the call is active	String	0: Asynchronous	Start Acc Stop Acc
Response Attributes						
26	h323-return-code	103	The reason for failing authentication (0 = ok, other number failed)	Numeric	0 Request accepted	Stop Acc
44	Acct-Session-Id		A unique accounting identifier – match start & stop	String		Stop Acc

9.11.1 RADIUS Server Messages

In Figure 9-5 below, non-standard parameters are preceded with brackets.

Figure 9-5: Accounting Example

```
Accounting-Request (361)
user-name = 111
acct-session-id = 1
nas-ip-address = 212.179.22.213
nas-port-type = 0
acct-status-type = 2
acct-input-octets = 4841
acct-output-octets = 8800
acct-session-time = 1
acct-input-packets = 122
acct-output-packets = 220
called-station-id = 201
calling-station-id = 202
// Accounting non-standard parameters:
(4923 33) h323-gw-id =
(4923 23) h323-remote-address = 212.179.22.214
(4923 1) h323-ivr-out = h323-incoming-conf-id:02102944 600a1899
3fd61009 0e2f3cc5
(4923 30) h323-disconnect-cause = 22 (0x16)
(4923 27) h323-call-type = VOIP
(4923 26) h323-call-origin = Originate
(4923 24) h323-conf-id = 02102944 600a1899 3fd61009 0e2f3cc5
```

9.12 Trunk-to-Trunk Routing Example

This example describes two Mediant 1000 gateways, each interfacing with the PSTN through four E1 spans. Gateway ‘A’ is configured to route all incoming Tel→IP calls to gateway ‘B’. Gateway ‘B’ generates calls to PSTN on the same E1 Trunk as the call was originally received (in gateway ‘A’).

- Gateway ‘A’ IP address: 192.168.3.50
- Gateway ‘B’ IP address: 192.168.3.51

***The Ini* file parameters configuration for gateways ‘A’ and ‘B’ include the following:**

1. Define, for both gateways, four trunk groups, each with 30 B-channels:
   - TrunkGroup_1 = 0/1-31,1000
   - TrunkGroup_2 = 1/1-31,2000
   - TrunkGroup_3 = 2/1-31,3000
   - TrunkGroup_4 = 3/1-31,4000
2. In gateway ‘A’, add the originating Trunk Group ID as a prefix to the destination number for Tel→IP calls:

   AddTrunkGroupAsPrefix = 1
3. In gateway ‘A’, route all incoming PSTN calls, starting with the prefixes 1, 2, 3, and 4, to gateway’s ‘B’ IP address:
   - Prefix = 1, 192.168.3.51
   - Prefix = 2, 192.168.3.51
   - Prefix = 3, 192.168.3.51
   - Prefix = 4, 192.168.3.51

   Note: It is also possible to define Prefix = *,192.168.3.51 instead of the four lines above.
4. In gateway ‘B’, route IP→PSTN calls to Trunk Group ID according to the first digit of the called number:
   - PSTNPrefix = 1,1
   - PSTNPrefix = 2,2
   - PSTNPrefix = 3,4
   - PSTNPrefix = 4,4
5. In gateway ‘B’, remove the first digit from each IP→PSTN number before it is used in an outgoing call:

   NumberMapIP2Tel = *,1

9.13 Proxy or Registrar Registration Example

The REGISTER message is sent to the Registrar's IP address (if configured) or to the Proxy's IP address. The message is sent per gateway or per gateway endpoint according to the 'AuthenticationMode' parameter. Usually the FXS modules are registered per gateway port, while FXO modules send a single registration message, where Username is used instead of phone number in From/To headers. The registration request is resent according to the parameter 'RegistrartionTimeDivider'. For example, if 'RegistrationTimeDivider = 70' (%) and Registration Expires time = 3600, the gateway resends its registration request after 3600 x 70% = 2520 sec. The default value of 'RegistrartionTimeDivider' is 50%.

If registration per B-channel is selected, on gateway startup, the gateway sends REGISTER requests according to the maximum number of allowed SIP dialogs (configured by the parameter NumberOfActiveDialogs). After each received response, the subsequent REGISTER request is sent.

```
REGISTER sip:servername SIP/2.0
VIA: SIP/2.0/UDP 212.179.22.229;branch=z9hG4bRaC7AU234
From: <sip:101@sipgatewayname>;tag=1c29347
To: <sip:101@sipgatewayname>
Call-ID: mailto:10453@212.179.22.229
Seq: 1 REGISTER
Expires: 3600
Contact: sip:101@212.179.22.229
Content-Length: 0
```

The 'servername' string is defined according to the following rules:

- The 'servername' is equal to 'RegistrarName' if configured. The 'RegistrarName' can be any string.
- Otherwise, the 'servername' is equal to 'RegistrarIP' (either FQDN or numerical IP address), if configured.
- Otherwise the 'servername' is equal to 'ProxyName' if configured. The 'ProxyName' can be any string.
- Otherwise the 'servername' is equal to 'ProxyIP' (either FQDN or numerical IP address).

The 'sipgatewayname' parameter (defined in the *ini* file or set from the Web browser -- SIPGatewayName), can be any string. Some Proxy servers require that the 'sipgatewayname' (in REGISTER messages) is set equal to the Registrar/Proxy IP address or to the Registrar/Proxy domain name.

9.14 Configuration Examples

9.14.1 Establishing a Call between Two Gateways

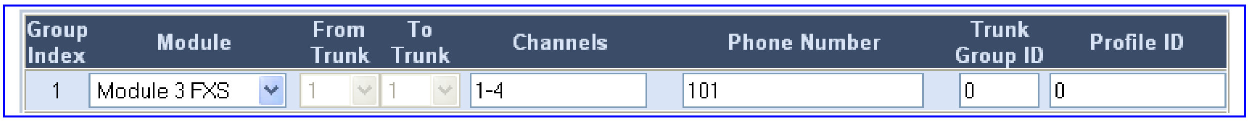
After you've installed and set up the Mediant 1000, you can ensure that it functions as expected by establishing a call between it and another gateway. This section describes how to configure two 4-port Mediant 1000 FXS SIP gateways to establish a call. After configuration, you can make calls between telephones connected to a single Mediant 1000 gateway or between the two Mediant 1000 gateways.

In the following example, the IP address of the first gateway is 10.2.37.10 and its endpoint numbers are 101 to 104. The IP address of the second gateway is 10.2.37.20 and its endpoint numbers are 201 to 204.

In this example, a SIP Proxy is not used. Internal call routing is performed using the internal 'Tel to IP Routing' table.

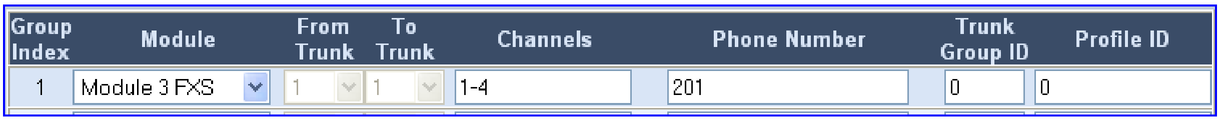
➢ To configure the two gateways, take these 4 steps:

1. For the *first* Mediant 1000 gateway (10.2.37.10), in the 'Trunk Group Table' screen (**Protocol Management** menu > **Trunk Group**), assign the phone numbers 101 to 104 for the gateway's endpoints.



Group Index	Module	From Trunk	To Trunk	Channels	Phone Number	Trunk Group ID	Profile ID
1	Module 3 FXS	1	1	1-4	101	0	0

2. For the *second* Mediant 1000 gateway (10.2.37.20), in the 'Trunk Group Table' screen, assign the phone numbers 201 to 204 for the gateway's endpoints.

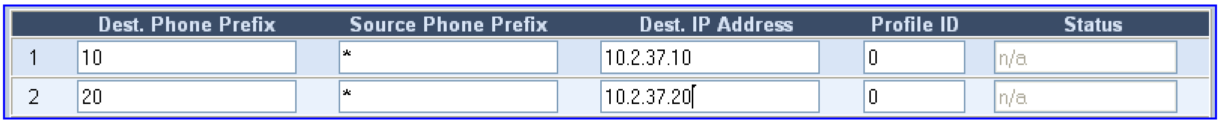


Group Index	Module	From Trunk	To Trunk	Channels	Phone Number	Trunk Group ID	Profile ID
1	Module 3 FXS	1	1	1-4	201	0	0

3. Configure the following settings for *both* gateways:

   In the 'Tel to IP Routing' screen (**Protocol Management** menu > **Routing Tables** submenu **> Tel to IP Routing**), in the first row, enter 10 in the 'Destination Phone Prefix' field and enter the IP address of the first gateway (10.2.37.10) in the field 'IP Address'. In the second row, enter 20 and the IP address of the second gateway (10.2.37.20) respectively.

   These settings enable the routing (from both gateways) of outgoing Tel→IP calls that start with 10 to the first gateway and calls that start with 20 to the second gateway.



	Dest. Phone Prefix	Source Phone Prefix	Dest. IP Address	Profile ID	Status
1	10	*	10.2.37.10	0	n/a
2	20	*	10.2.37.20	0	n/a

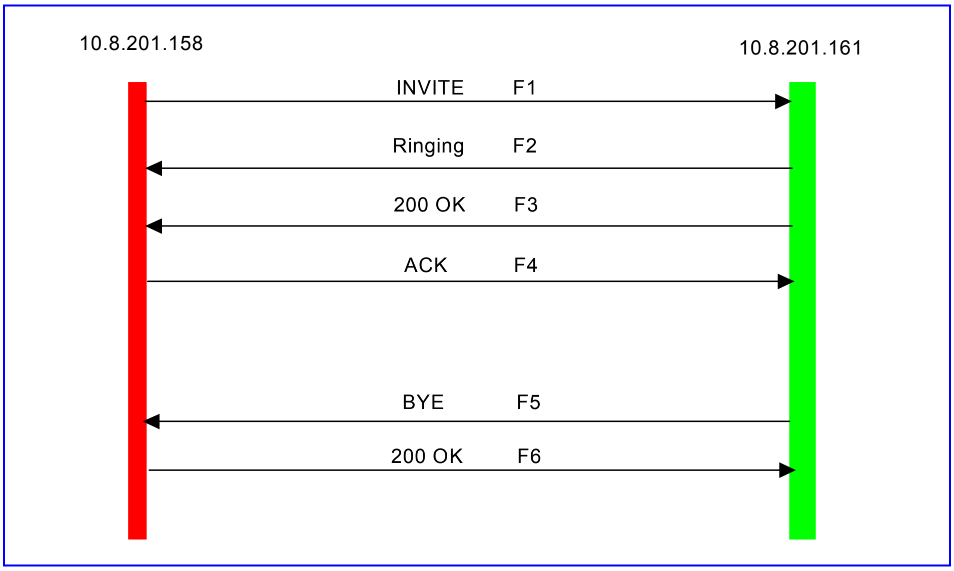
4. Make a call. Pick up the phone connected to port #1 of the first Mediant 1000 and dial 102 (to the phone connected to port #2 of the same gateway). Listen for progress tones at the calling endpoint and for ringing tone at the called endpoint. Answer the called endpoint, speak into the calling endpoint, and check the voice quality. Dial 201 from the phone connected to port #1 of the first Mediant 1000 gateway; the phone connected to port #1 of the second Mediant 1000 rings. Answer the call and check the voice quality.

9.14.2 SIP Call Flow

The following Call Flow describes SIP messages exchanged between two Mediant 1000 gateways during simple call.

Phone ‘6000’ dials ‘2000’, sending INVITE message to Gateway 10.8.201.161

Figure 9-6: SIP Call Flow



F1 10.8.201.158 ==> 10.8.201.161 INVITE

```
INVITE sip:6000@10.8.201.161;user=phone SIP/2.0
Via: SIP/2.0/UDP 10.8.201.158;branch=z9hG4bKacolwbzYF
From: <sip:2000@10.8.201.158>;tag=1c3535
To: <sip:6000@10.8.201.161>
Call-ID: 2123353775377NrpL-2000--6000@10.8.201.158
CSeq: 20214 INVITE
Contact: <sip:2000@10.8.201.158;user=phone>
User-Agent: Audiocodes-Sip-Gateway/Mediant 1000/v.4.20.299.410
Supported: 100rel,em
Allow: REGISTER,OPTIONS,INVITE,ACK,CANCEL,BYE,NOTIFY,PRACK,REFER,INFO
Content-Type: application/sdp
Content-Length: 208

v=0
s=Phone-Call
t=0 0
o=AudiocodesGW 87943 43401 IN IP4 10.8.201.158
c=IN IP4 10.8.201.158
m=audio 6000 RTP/AVP 8 96
a=rtpmap:8 pcma/8000
a=rtpmap:96 telephone-event/8000
a=fmtp:96 0-15
a=ptime:20
```

F2 10.8.201.161 ==> 10.8.201.158 180 RINGING

```
SIP/2.0 180 Ringing
Via: SIP/2.0/UDP 10.8.201.158;branch=z9hG4bKacolwbzYF
From: <sip:2000@10.8.201.158>;tag=1c3535
To: <sip:6000@10.8.201.161>;tag=1c29715
Call-ID: 2123353775377NrpL-2000--6000@10.8.201.158
Server: Audiocodes-Sip-Gateway/Mediant 1000/v.4.20.299.410
CSeq: 20214 INVITE
Supported: 100rel,em
Content-Length: 0
```

Note: Phone ‘2000’ answers the call, and sends 200 OK message to gateway 10.8.201.158.

F3 10.8.201.161 ==> 10.8.201.158 200 OK

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP 10.8.201.158;branch=z9hG4bKacolwbzYF
From: <sip:2000@10.8.201.158>;tag=1c3535
To: <sip:6000@10.8.201.161>;tag=1c29715
Call-ID: 2123353775377NrpL-2000--6000@10.8.201.158
CSeq: 20214 INVITE
Contact: <sip:6000@10.8.201.161;user=phone>
Server: Audiocodes-Sip-Gateway/Mediant 1000/v.4.20.299.410
Supported: 100rel,em
Allow: REGISTER,OPTIONS,INVITE,ACK,CANCEL,BYE,NOTIFY,PRACK,REFER,INFO
Content-Type: application/sdp
Content-Length: 208

v=0
s=Phone-Call
t=0 0
o=AudiocodesGW 30762 37542 IN IP4 10.8.201.161
c=IN IP4 10.8.201.161
m=audio 4040 RTP/AVP 8 96
a=rtpmap:8 pcma/8000
a=ptime:20
a=rtpmap:96 telephone-event/8000
a=fmtp:96 0-15
```

F4 10.8.201.158 ==> 10.8.201.161 ACK

```
ACK sip:6000@10.8.201.161;user=phone;user=phone SIP/2.0
Via: SIP/2.0/UDP 10.8.201.158;branch=z9hG4bKachoWSQxD
From: <sip:2000@10.8.201.158>;tag=1c3535
To: <sip:6000@10.8.201.161>;tag=1c29715
Call-ID: 2123353775377NrpL-2000--6000@10.8.201.158
User-Agent: Audiocodes-Sip-Gateway/Mediant 1000/v.4.20.299.410
CSeq: 20214 ACK
Supported: 100rel,em
Content-Length: 0
```

Note: Phone ‘6000’ goes onhook, gateway 10.8.201.161 sends BYE to gateway 10.8.201.158. Voice path is established.

F5 10.8.201.161 ==> 10.8.201.158 BYE

```
BYE sip:2000@10.8.201.158;user=phone;user=phone SIP/2.0
Via: SIP/2.0/UDP 10.8.201.161;branch=z9hG4bKacLBzZgmA
From: <sip:6000@10.8.201.161>;tag=1c29715
To: <sip:2000@10.8.201.158>;tag=1c3535
Call-ID: 2123353775377NrpL-2000--6000@10.8.201.158
User-Agent: Audiocodes-Sip-Gateway/Mediant 1000/v.4.20.299.410
CSeq: 34541 BYE
Supported: 100rel,em
Content-Length: 0
```

F6 10.8.201.158 ==> 10.8.201.161 200 OK

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP 10.8.201.161;branch=z9hG4bKacLBzZgmA
From: <sip:6000@10.8.201.161>;tag=1c29715
To: <sip:2000@10.8.201.158>;tag=1c3535
Call-ID: 2123353775377NrpL-2000--6000@10.8.201.158
Server: Audiocodes-Sip-Gateway/Mediant 1000/v.4.20.299.410
CSeq: 34541 BYE
Supported: 100rel,em
Content-Length: 0
```

9.14.3 SIP Authentication Example

Mediant 1000 gateways support basic and digest (MD5) authentication types, according to SIP RFC 3261 standard. A proxy server might require authentication before forwarding an INVITE message. A Registrar/Proxy server may also require authentication for client registration. A proxy replies to an unauthenticated INVITE with a 407 Proxy Authorization Required response, containing a Proxy-Authenticate header with the form of the challenge. After sending an ACK for the 407, the user agent can then resend the INVITE with a Proxy-Authorization header containing the credentials.

User agent, redirect or registrar servers typically use 401 Unauthorized response to challenge authentication containing a WWW-Authenticate header, and expect the re-INVITE to contain an Authorization header.

The following example describes the Digest Authentication procedure including computation of user agent credentials.

The REGISTER request is sent to Registrar/Proxy server for registration, as follows:

```
REGISTER sip:10.2.2.222 SIP/2.0
Via: SIP/2.0/UDP 10.1.1.200
From: <sip: 122@10.1.1.200>;tag=1c17940
To: <sip: 122@10.1.1.200>
Call-ID: 634293194@10.1.1.200
User-Agent: Audiocodes-Sip-Gateway/Mediant 1000/v.4.20.299.410
CSeq: 1 REGISTER
Contact: sip:122@10.1.1.200:
Expires:3600
```

On receiving this request the Registrar/Proxy returns 401 Unauthorized response.

```
SIP/2.0 401 Unauthorized
Via: SIP/2.0/UDP 10.2.1.200
From: <sip:122@10.2.2.222 >;tag=1c17940
To: <sip:122@10.2.2.222 >
Call-ID: 634293194@10.1.1.200
Cseq: 1 REGISTER
Date: Mon, 30 Jul 2001 15:33:54 GMT
Server: Columbia-SIP-Server/1.17
Content-Length: 0
WWW-Authenticate: Digest realm="audiocodes.com",
nonce="11432d6bce58ddf02e3b5e1c77c010d2",
stale=FALSE,
algorithm=MD5
```

According to the sub-header present in the WWW-Authenticate header the correct REGISTER request is formed.

Since the algorithm used is MD5, take:

The username is equal to the endpoint phone number: 122

The realm return by the proxy: audiocodes.com

The password from the ini file: AudioCodes.

The equation to be evaluated: (according to RFC this part is called A1).

‘122:audiocodes.com:AudioCodes’.

The MD5 algorithm is run on this equation and stored for future usage.

The result is: ‘a8f17d4b41ab8dab6c95d3c14e34a9e1’

Next we need to evaluate the par called A2. We take:

The method type ‘REGISTER’

Using SIP protocol ‘sip’

Proxy IP from *ini* file ‘10.2.2.222’

The equation to be evaluated:

‘REGISTER:sip:10.2.2.222’.

The MD5 algorithm is run on this equation and stored for future usage.

The result is:’a9a031cfddcb10d91c8e7b4926086f7e’

The final stage:

The A1 result

The nonce from the proxy response: ‘11432d6bce58ddf02e3b5e1c77c010d2’

The A2 result

The equation to be evaluated:

‘A1:11432d6bce58ddf02e3b5e1c77c010d2:A2’.

The MD5 algorithm is run on this equation. The outcome of the calculation is the response needed by the gateway to be able to register with the Proxy.

The response is: ‘b9c45d0234a5abf5ddf5c704029b38cf’

At this time a new REGISTER request is issued with the response:

```
REGISTER sip:10.2.2.222 SIP/2.0
Via: SIP/2.0/UDP 10.1.1.200
From: <sip: 122@10.1.1.200>;tag=1c23940
To: <sip: 122@10.1.1.200>
Call-ID: 654982194@10.1.1.200
Server: Audiocodes-Sip-Gateway/Mediant 1000/v.4.20.299.410
CSeq: 1 REGISTER
Contact: sip:122@10.1.1.200:
Expires:3600
Authorization: Digest, username: 122,
realm="audiocodes.com”,
nonce="11432d6bce58ddf02e3b5e1c77c010d2",
uri=”10.2.2.222”,
response=“b9c45d0234a5abf5ddf5c704029b38cf”
```

On receiving this request, if accepted by the Proxy, the proxy returns a 200 OK response closing the REGISTER transaction.

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP 10.1.1.200
From: <sip: 122@10.1.1.200>;tag=1c23940
To: <sip: 122@10.1.1.200>
Call-ID: 654982194@10.1.1.200
Cseq: 1 REGISTER
Date: Thu, 26 Jul 2001 09:34:42 GMT
Server: Columbia-SIP-Server/1.17
Content-Length: 0
Contact: <sip:122@10.1.1.200>; expires="Thu, 26 Jul 2001 10:34:42 GMT";
action=proxy; q=1.00
Contact: <122@10.1.1.200:>; expires="Tue, 19 Jan 2038 03:14:07 GMT";
action=proxy; q=0.00
Expires: Thu, 26 Jul 2001 10:34:42 GMT
```

9.14.4 Remote IP Extension between FXO and FXS

This application explains how to implement remote extension via IP, using 4-port FXO and 4-port FXS Mediant 1000 gateways. In this configuration, PBX incoming calls are routed to the ‘Remote Extension’ via the FXO and FXS modules.

Requirements:

- One FXO Mediant 1000 gateway
- One FXS Mediant 1000 gateway
- Analog phones (POTS)
- PBX – one or more PBX loop start lines
- LAN

Connect the FXO Mediant 1000 ports directly to the PBX lines.

9.14.4.1 Dialing from Remote Extension (Phone Connected to FXS)

➢ To configure the call, take these 3 steps:

1. Lift the handset to hear the dial tone coming from PBX, as if the phone was connected directly to PBX. The FXS and FXO Mediant 1000 gateways establish a voice path connection from the phone to the PBX immediately after the phone handset is raised.
2. Dial the destination number (the DTMF digits are sent, over IP, directly to the PBX). All tones heard are generated from the PBX (such as Ringback, busy or fast busy tones). There is one-to-one mapping between FXS ports and PBX lines.
3. The call is disconnected when the phone connected to the FXS goes onhook.

9.14.4.2 Dialing from other PBX line, or from PSTN

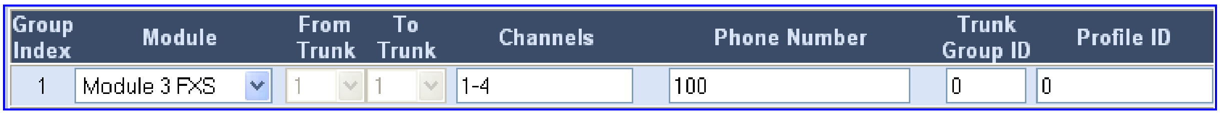
➢ **To configure the call, take these 5 steps:**

1. Dial the PBX subscriber number the same way as if the user's phone was connected directly to PBX.
2. Immediately as PBX rings into FXO Mediant 1000, the ring signal is 'sent' to phone connected to FXS Mediant 1000.
3. Once the phone's handset, connected to FXS Mediant 1000, is raised, the FXO Mediant 1000 seizes the PBX line and the voice path is established between the phone and the PBX line.
4. There is a one to one mapping between PBX lines and FXS Mediant 1000 ports. Each PBX line is routed to the same phone (connected to FXS Mediant 1000).
5. The call is disconnected when phone connected to FXS Mediant 1000 goes onhook.

9.14.4.3 FXS Mediant 1000 Configuration (using the Embedded Web Server)

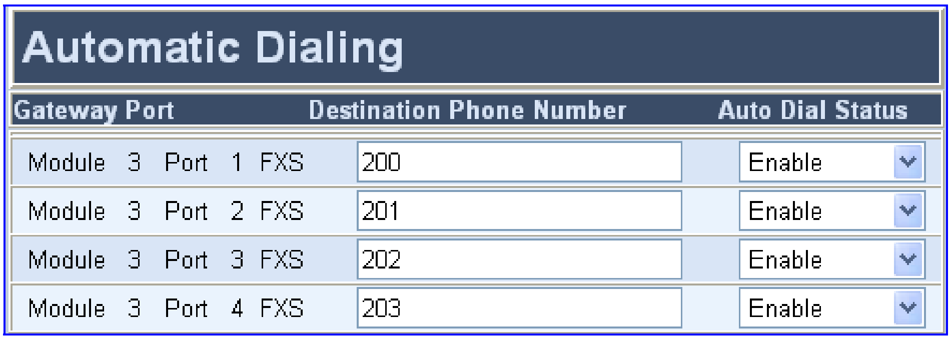
➢ **To configure the FXS Mediant 1000, take these 3 steps:**

1. In the 'Endpoint Phone Numbers' screen, assign the phone numbers 100 to 107 for the gateway's endpoints.



Group Index	Module	From Trunk	To Trunk	Channels	Phone Number	Trunk Group ID	Profile ID
1	Module 3 FXS	1	1	1-4	100	0	0

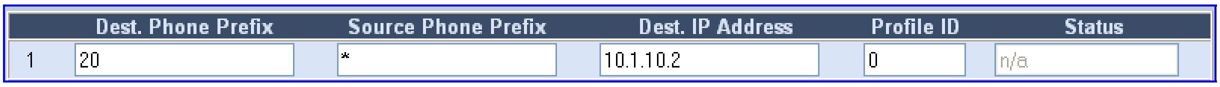
2. In the 'Automatic Dialing' screen, enter the phone numbers of the FXO Mediant 1000 gateway in the 'Destination Phone Number' fields. When a phone connected to port #1 goes offhook, the FXS module automatically dials the number '200'.



Automatic Dialing

Gateway Port	Destination Phone Number	Auto Dial Status
Module 3 Port 1 FXS	200	Enable
Module 3 Port 2 FXS	201	Enable
Module 3 Port 3 FXS	202	Enable
Module 3 Port 4 FXS	203	Enable

3. In the 'Tel to IP Routing' screen, enter 20 in the 'Destination Phone Prefix' field, and the IP address of the FXO Mediant 1000 gateway (10.1.10.2) in the field 'IP Address'.



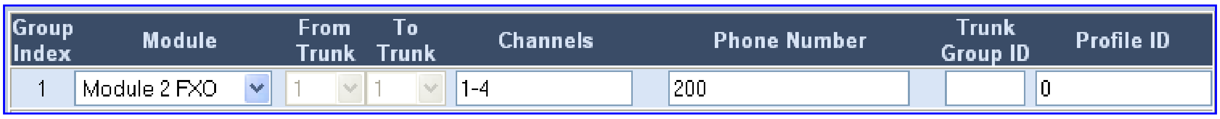
	Dest. Phone Prefix	Source Phone Prefix	Dest. IP Address	Profile ID	Status
1	20	*	10.1.10.2	0	n/a

Note: In remote extensions, for the transfer to function, hold must be disabled on the FXS (i.e., Enable Hold = 0).

9.14.4.4 FXO Mediant 1000 Configuration (using the Embedded Web Server)

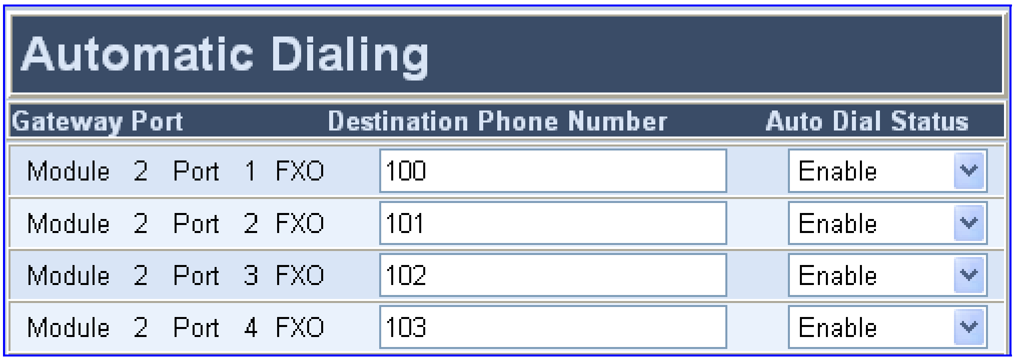
➢ To configure the FXO Mediant 1000, take these 4 steps:

1. In the 'Endpoint Phone Numbers' screen, assign the phone numbers 200 to 207 for the gateway's endpoints.



Group Index	Module	From Trunk	To Trunk	Channels	Phone Number	Trunk Group ID	Profile ID
1	Module 2 FXO	1	1	1-4	200		0

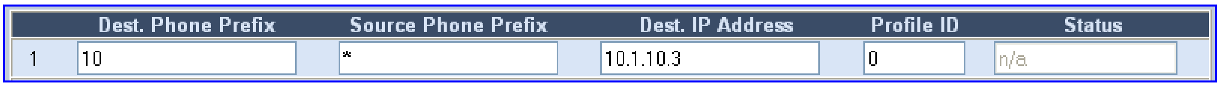
2. In the 'Automatic Dialing' screen, enter the phone numbers of the FXS Mediant 1000 gateway in the 'Destination Phone Number' fields. When a ringing signal is detected at port #1, the FXO module automatically dials the number '100'.



Automatic Dialing

Gateway Port	Destination Phone Number	Auto Dial Status
Module 2 Port 1 FXO	100	Enable
Module 2 Port 2 FXO	101	Enable
Module 2 Port 3 FXO	102	Enable
Module 2 Port 4 FXO	103	Enable

3. In the 'Tel to IP Routing' screen, enter 10 in the 'Destination Phone Prefix' field, and the IP address of the FXS Mediant 1000 gateway (10.1.10.3) in the field 'IP Address'.



	Dest. Phone Prefix	Source Phone Prefix	Dest. IP Address	Profile ID	Status
1	10	*	10.1.10.3	0	n/a

4. In the 'Protocol Management' screen, set the parameter 'Dialing Mode' to 'Two Stage' (IsTwoStageDial=1).

Reader's Notes

10 Networking Capabilities

10.1 Ethernet Interface Configuration

Using the parameter ‘EthernetPhyConfiguration‘, users can control the Ethernet connection mode.

Either the manual modes (10 Base-T Half-Duplex, 10 Base-T Full-Duplex, 100 Base-TX Half-Duplex, 100 Base-TX Full-Duplex) or Auto-Negotiate mode can be used.

Auto-Negotiation falls back to Half-Duplex mode when the opposite port is not Auto-Negotiate, but the speed (10 Base-T, 100 Base-TX) in this mode is always configured correctly. Note that configuring the gateway to Auto-Negotiate mode while the opposite port is set manually to Full-Duplex (either 10 Base-T or 100 Base-TX) is invalid (as it causes the gateway to fall back to Half-Duplex mode while the opposite port is Full-Duplex). It is also invalid to set the gateway to one of the manual modes while the opposite port is either Auto-Negotiate or not exactly matching (both in speed and in duplex mode). Users are encouraged to always prefer Full-Duplex connections to Half-Duplex ones and 100 Base-TX to 10 Base-T (due to the larger bandwidth). It is strongly recommended to use the same mode in both link partners. Any mismatch configuration can yield unexpected functioning of the Ethernet connection.

Note that when remote configuration is performed, the gateway should be in the correct Ethernet setting prior to the time this parameter takes effect. When, for example, the gateway is configured using BootP/TFTP, the gateway must perform many Ethernet-based transactions prior to reading the *ini* file containing this gateway configuration parameter.

To work around this problem, the gateway always uses the last Ethernet setup mode configured. This way, if users want to configure the gateway to work in a new network environment in which the current Ethernet setting of the gateway is invalid, they should first modify this parameter in the current network so that the new setting holds next time gateway is restarted. After reconfiguration has completed, connect the gateway to the new network and restart it. As a result, the remote configuration process that takes place in the new network uses a valid Ethernet configuration.

10.2 Ethernet Interface Redundancy

The Mediant 1000 supports Ethernet interface redundancy by providing two 10/100 Base-TX Ethernet ports and supporting Ethernet switchover mechanisms. The 'MIIRedundancyEnable' parameter (refer to Section 6.5.1 on page 215) is used to enable Ethernet interface redundancy.

The Ethernet interface redundancy scheme functions as follows:

1. At the beginning of the start-up procedure, the gateway tests whether the ‘Primary’ Ethernet interface is connected, by checking the existence of the Ethernet link carrier. If it is connected, the start-up procedure commences as usual. If not, the start-up application tries the ‘Secondary’ Ethernet interface. If this interface is connected, the entire start-up procedure is performed using this ‘Secondary’ Ethernet interface. If both interfaces are out of service (i.e. failure), the start-up procedure commences using the parameters, tables, and software residing on the gateway’s non-volatile memory.

Note: Ethernet switchover occurs only once during the start-up procedure (at the beginning). If the Ethernet interface fails after selecting an Ethernet port, the gateway doesn't switch over to the second port.

2. After start-up has completed and the operational software is running, the gateway continues to use the Ethernet port used for program load. The gateway switches over from one Ethernet port to the other every time an Ethernet link carrier loss is detected on the active Ethernet port and if the Ethernet link of the other port is operational. Switchover occurs only once per link loss (that is, the 'secondary' interface remains the active one even if the 'primary' interface has returned to "life"). In addition, the gateway generates a gratuitous ARP message each time a switchover occurs.

For correct functionality of the redundancy mechanism, it is recommended to configure the same mode for both links. It is essential that both link partners (the primary link partner and the secondary link partner) have the same capabilities, so that whenever a switchover occurs the gateway is able to provide at least the same Ethernet services as were provided prior to the switchover. In addition, it is recommended to configure the physical secondary link prior to the gateway being reset (since the MAC configuration cannot be changed thereafter).

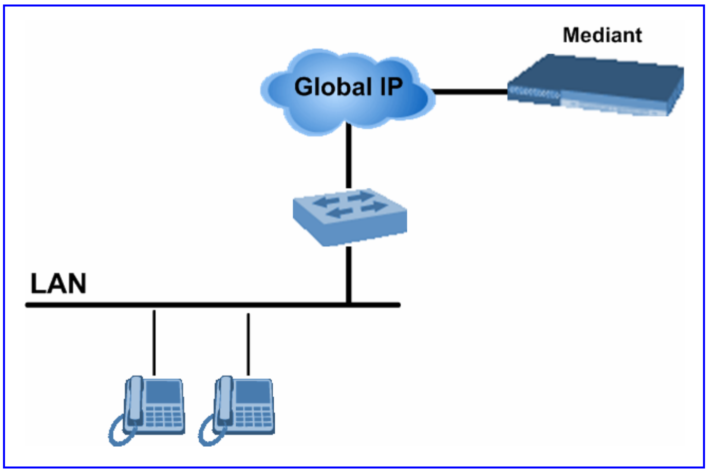
Note: Since the two Ethernet ports use the same MAC address, the external switches connected to the gateways can in some cases create a noticeable switchover delay due to their internal switching logic, though at the gateway level, the switchover delay is minimal (milliseconds).

10.3 NAT (Network Address Translation) Support

NAT is a mechanism that maps a set of internal IP addresses used within a private network to global IP addresses, providing transparent routing to end hosts. The primary advantages of NAT are (1) Reduction of the number of global IP addresses required in a private network (global IP addresses are only used to connect to the Internet); (2) Better network security by hiding its internal architecture.

Figure 10-1 illustrates the NAT architecture.

Figure 10-1: NAT Functioning



The way SIP is designed creates a problem for VoIP traffic to pass through NAT. SIP uses IP addresses and port numbers in its message body. The NAT server can't modify SIP messages and therefore, can't change local to global addresses.

Two different streams traverse through NAT: signaling and media. A gateway (located behind a NAT) that initiates a signaling path will have problems in receiving incoming signaling responses (they will be blocked by the NAT). Furthermore, the initiating gateway must notify the receiving gateway where to send the media to.

To solve these problems the following mechanisms are available:

- STUN (refer to Section 10.3.1 below).
- First Incoming Packet Mechanism (refer to Section 10.3.2 on page 304)
- RTP / T.38 No-Op packets according to the avt-rtp-noop draft (refer to Section 10.3.3 on page 304).
- For SNMP NAT traversal, refer to Section 16.10 on page 388.

10.3.1 STUN

Simple Traversal of UDP through NATs (STUN) (according to RFC 3489) is a client / server protocol that solves most of the NAT traversal problems. The STUN server operates in the public Internet and the STUN clients are embedded in end-devices (located behind NAT). STUN is used both for the signaling and the media streams. STUN works with many existing NAT types, and does not require any special behavior from them.

STUN enables the gateway to discover the presence (and types) of NATs and firewalls located between it and the public Internet. It provides the gateway with the capability to determine the public IP address and port allocated to it by the NAT. This information is later embedded in outgoing SIP/SDP messages and enables remote SIP user agents to reach the gateway. It also discovers the binding lifetime of the NAT (the refresh rate necessary to keep NAT 'Pinholes' open).

On startup the gateway sends a STUN Binding Request. The information received in the STUN Binding Response (IP address:port) is used for SIP signaling. This information is updated every NATBindingDefaultTimeout.

At the beginning of each call, if STUN is needed (i.e., not an internal NAT call), the media ports of the call are mapped. The call is delayed until the STUN Binding Response (that includes a global IP:port) for each media (RTP, RTCP and T.38) is received.

To enable STUN:

- Set the parameter EnableSTUN to 1
- Define the STUN server address using one of the following methods:
  - Define the IP address of the primary and the secondary (optional) STUN servers using the parameters STUNServerPrimaryIP and STUNServerSecondaryIP. If the primary STUN server isn't available, the gateway tries to communicate with the secondary server.
  - Define the domain name of the STUN server using the StunServerDomainName parameter. The STUN client retrieves all STUN servers with an SRV query to resolve this domain name to an IP address and port, sort the server list, and use the servers according to the sorted list.
- Use the parameter NATBindingDefaultTimeout to determine the default NAT binding lifetime in seconds. STUN is used to refresh the binding information after this time expires.

Notes:

- STUN only applies to UDP (doesn't support TCP and TLS).
- STUN can't be used when the gateway is located behind a symmetric NAT.
- For defining the STUN server, use either the STUNServerPrimaryIP or STUNServerDomainName parameter, with priority to the first one.

10.3.2 First Incoming Packet Mechanism

If the remote gateway resides behind a NAT device, it's possible that the Mediant 1000 can activate the RTP/RTCP/T.38 streams to an invalid IP address / UDP port. To avoid such cases, the Mediant 1000 automatically compares the source address of the incoming RTP/RTCP/T.38 stream with the IP address and UDP port of the remote gateway. If the two are not identical, the transmitter modifies the sending address to correspond with the address of the incoming stream. The RTP, RTCP and T.38 can thus have independent destination IP addresses and UDP ports.

Users can choose to disable the NAT mechanism by setting the *ini* file parameter 'DisableNAT' to 1. The two parameters 'EnableIpAddrTranslation' and 'EnableUdpPortTranslation' enable users to specify the type of compare operation that takes place on the first incoming packet. To compare only the IP address, set 'EnableIpAddrTranslation = 1' and 'EnableUdpPortTranslation = 0'. In this case, if the first incoming packet arrives with only a difference in the UDP port, the sending addresses won't change. If both the IP address and UDP port need to be compared, then both parameters need to be set to 1.

10.3.3 RTP / T.38 No-Op Packets

The gateway's No-Op packet support can be used to verify Real-Time Transport Protocol (RTP) and T.38 connectivity, and to keep NAT bindings and Firewall pinholes open. No-Op packets are available for sending in RTP and T.38 formats.

Users can control the activation of No-Op packets by using the *ini* file parameter NoOperationSendingMode. If No-Op packet transmission is activated, users can control the time interval in which No-Op packets are sent in the case of silence (i.e., no RTP or T.38 traffic). This is performed using the NoOpInterval *ini* parameter.

Note: Receipt of No-Op is always supported.

- **RTP No-Op:**

  The RTP No-Op support complies with IETF's draft-wing-avt-rtp-noop-03.txt (titled 'A No-Op Payload Format for RTP'). This IETF document defines a No-Op payload format for RTP.

  The draft defines the RTP payload type as dynamic. Users can control the payload type with which the No-Op packets are sent. This is performed using the RTPNoOpPayloadType *ini* parameter. AudioCodes' default payload type is 120.

- **T.38 No-Op:**

  T.38 No-Op packets are sent only while a T.38 session is activated. Sent packets are a duplication of the previously sent frame (including duplication of the sequence number).

10.4 Point-to-Point Protocol over Ethernet (PPPoE)

PPPoE is a method of sending the Point-to-Point Protocol over Ethernet network.

10.4.1 Point-to-Point Protocol (PPP) Overview

Point-to-Point Protocol (PPP) provides a method of transmitting data over serial point-to-point links. The protocol defines establishing, configuring and testing the data link connection and the network protocol.

The PPP standard describes a state machine used to establish a valid connection between two hosts over a serial connection. There are three major stages described, helping to establish a network layer (such as an IP) connection over the point-to-point link: LCP (Link Configuration Protocol), Authentication, and NCP (Network Control Protocol). Once the network protocol is configured, the two hosts can communicate, sending network layer protocol (such as IP) over the PPP connection (a small PPP header is added at the beginning of each packet).

At the initial phase, the hosts use LCP (link configuration protocol) to negotiate for link characteristic and parameters. Packets sent in this phase have two octets of ‘PPP header’ followed by LCP message with variable length. Various parameters and options are negotiable at this phase, including MRU (maximum receive unit), Authentication Protocol, and others.

Once the link is established (each side sends a ‘configure ack’ message to the other side), the authentication phase may begin. The authentication phase is not mandatory. However, it is negotiated in the link configuration phase. A host may ask other hosts for authentication using Password Authentication Protocol (PAP) or Challenge Handshake Authentication Protocol (CHAP).

The PAP sends the username and password to the remote host unencrypted.

The CHAP is a more sophisticated method of authentication. The two hosts share a ‘secret’. The authenticator sends a ‘challenge’ to the host requesting authentication. The host performs a calculation (one-way hash) using the challenge received from the authenticator and the shared ‘secret’, and sends the result to the authenticator. The authenticator verifies the host if the result of the calculation is correct; otherwise it is rejected.

The last configuration phase, immediately after the authentication phase (or after the Link Configuration) is the Network Control Protocol. There is a family of control protocols for establishing and configuring different network-layer protocols, for example, IPCP (PPP Internet Protocol Control Protocol), IPv6CP (PPP IP v6 Control Protocol), and BCP (PPP Bridging Control Protocol). Each of them handles and manages the specific needs required by their respective network-layer protocol.

When working in an IP network, IPCP is used as the Network Configuration Protocol. The IPCP is used to configure the network layer of the hosts, requesting/declaring on IP Addresses.

Further information on PPP Protocol is available on the IETF website (http://www.ietf.org/rfc/rfc1661.txt). Further information on Password Authentication Protocol is available on the IETF website (http://www.ietf.org/rfc/rfc1334.txt). Further information on Challenge Handshake Authentication Protocol is available on the IETF website (http://www.ietf.org/rfc/rfc1994.txt). Further information on PPP Internet Protocol Control Protocol (IPCP) is available on the IETF website (http://www.ietf.org/rfc/rfc1332.txt).

10.4.2 PPPoE Overview

PPPoE is a method of sending the Point-to-Point Protocol over Ethernet network. PPPoE provides the ability to connect a network of hosts over a simple bridging access device to a remote Access Concentrator. Access control, billing and type of service can be done on a per-user, rather than a per-site, basis.

A common use of the PPPoE is in the ADSL market: The home PC is connected to a modem via Ethernet, and the PC uses the PPPoE to ‘simulate’ as if it was directly connected to the remote host on a point-to-point connection.

Since PPPoE frames are sent over Ethernet, each PPP session must learn the Ethernet address of the remote peer, as well as establish a unique session identifier. The PPPoE standard describes a discovery protocol that provides this. A PPPoE session begins with a discovery phase. Only after this discovery is completed can the PPP state machine start (with LCP, Authentication etc, as described above).

Each of the Ethernet frames carrying PPP session has a standard Ethernet header followed by PPPoE header, and is sent with the remote host Ethernet MAC address (except for the very first one, in the discovery phase, which is broadcasted to all hosts).

Further information on the transmission of PPPoE is available on the IETF website (http://www.ietf.org/rfc/rfc2516.txt).

10.4.3 PPPoE in AudioCodes Gateways

The AudioCodes gateway contains a PPPoE client embedded in its software. When correctly configured (see *ini* file parameters) the gateway can try to connect to a remote PPPoE Access Concentrator.

When restarting the gateway after several BOOTP attempts, if PPPoE is enabled (see *ini* file parameter EnablePPPoE), the gateway tries to initiate a PPP session.

The gateway initiates a PPPoE discovery phase to discover a PPPoE Access Concentrator. It does this by broadcasting a discovery initialization packet (PADI). If an Access Concentrator exists and replies, the gateway tries to connect to this Access Concentrator. If this initial connection succeeds, then the PPP LCP phase starts - each side of the PPPoE connection sends LCP configuration requests to configure the PPP link.

The gateway PPPoE client supports both PAP and CHAP authentications. The type of authentication protocol used is according to the request from the authentication server. In the LCP configuration phase, the server requires a specific authentication (none, PAP, or CHAP are supported). The *ini* file parameters PPPoEUserName, PPPoEPassword, and PPPoEServerName are used to configure the authentication parameters. If the Access Concentrator is configured to operate in PAP, the PPPoEUserName and PPPoEPassword are used as Username and Password (in this case, the PPPoEServerName parameter is not used). If the Access Concentrator is configured to operate in CHAP, the PPPoEUserName parameter functions as Client Name (sent in the CHAP response packet), while the PPPoEPassword functions as the shared secret (calculated along with the challenge to produce the response). In this case, the PPPoEServerName is the name of the server. Some hosts can be configured to authenticate to multiple servers. In such hosts, the server name is used to identify the "secret" that should be used.

Note: The AudioCodes gateway, being a PPPoE client, requests no authentication.

After the gateway has been authenticated, it needs to configure a network layer protocol. The gateway uses the IP protocol. Therefore, the used NCP will be IPCP (IP Configuration Protocol). In this phase, if the *ini* file parameter PPPoEStaticIPAddress is defined, the gateway requests the remote host to assign this address for its use.

When working in a PPPoE environment, the gateway negotiates for its IP address (as described above). However, if the user desires to disable the PPPoE client, the gateway can be configured to use default values for IP address, subnet mask and default gateway. This can be done using *ini* file parameters PPPoERecoverIPAddress, PPPoERecoverSubnetMask and PPPoERecoverDfgwAddress. These parameters indicate to the gateway that if the PPPoE is disabled and no BootP server is activated, as required in the gateway to use a PPPoE environment, then the gateway should use these defaults for its IP configuration.

For a detailed description of the *ini* file parameters for PPPoE, refer to Section 6.5.1 on page 215.

Note: When working with a PPPoE server (Access Concentrator) that does not reply to LCP Echo messages (which by default, the gateway periodically sends) you may want to disable the LCP Echo messages by using the *ini* file parameter PPPoELCPEchoEnable. (For a description of this parameter, refer to Section 6.5.1 on page 215.)

10.5 IP Multicasting

The gateway supports IP Multicasting level 1 according to RFC 2236 (i.e. IGMP version 2) for RTP channels. The gateway is capable of transmitting and receiving Multicast packets.

10.6 Robust Reception of RTP Streams

This mechanism filters out unwanted RTP streams that are sent to the same port number on the gateway. These multiple RTP streams can result from traces of previous calls, call control errors and deliberate attacks.

When more than one RTP stream reaches the gateway on the same port number, the gateway accepts only one of the RTP streams and rejects the rest of the streams. The RTP stream is selected according to the following procedure:

The first packet arriving on a newly opened channel sets the source IP address and UDP port from which further packets are received. Thus, the source IP address and UDP port identify the currently accepted stream. If a new packet arrives whose source IP address or UDP port are different to the currently accepted RTP stream, there are two options:

- The new packet has a source IP address and UDP port which are the same as the remote IP address and UDP port that were stated during the opening of the channel. In this case, the gateway reverts to this new RTP stream.
- The new packet has any other source IP address and UDP port, in which case the packet is dropped.

10.7 Multiple Routers Support

Multiple routers support is designed to assist the media gateway when it operates in a multiple routers network. The gateway learns the network topology by responding to ICMP redirections and caches them as routing rules (with expiration time).

When a set of routers operating within the same subnet serve as gateways to that network and intercommunicate using a dynamic routing protocol, the routers can determine the shortest path to a certain destination and signal the remote host the existence of the better route. Using multiple router support the media gateway can utilize these router messages to change its next hop and establish the best path.

Note: Multiple Routers support is an integral feature that doesn’t require configuration.

10.8 Simple Network Time Protocol Support

Simple Network Time Protocol (SNTP) client functionality generates requests and reacts to the resulting responses using the NTP version 3 protocol definitions (according to RFC 1305). Through these requests and responses, the NTP client is able to synchronize the system time to a time source within the network, thereby eliminating any potential issues should the local system clock 'drift' during operation. By synchronizing time to a network time source, traffic handling, maintenance, and debugging actions become simplified for the network administrator.

The NTP client follows a simple process in managing system time; the NTP client requests an NTP update, receives an NTP response, and updates the local system clock based on a configured NTP server within the network.

The client requests a time update from a specified NTP server at a specified update interval. In most situations this update interval should be every 24 hours based on when the system was restarted. The NTP server identity (as an IP address) and the update interval are configurable parameters that can be specified either in the *ini* file (NTPServerIP, NTPUpdateInterval respectively) or via an SNMP MIB object.

When the client receives a response to its request from the identified NTP server it must be interpreted based on time zone, or location, offset that the system is to a standard point of reference called the Universal Time Coordinate (UTC). The time offset that the NTP client should use is a configurable parameter that can be specified either in the *ini* file (NTPServerUTCOffset) or via an SNMP MIB object.

If required, the clock update is performed by the client as the final step of the update process. The update is done in such a way as to be transparent to the end users. For instance, the response of the server may indicate that the clock is running too fast on the client. The client slowly robs bits from the clock counter in order to update the clock to the correct time. If the clock is running too slow, then in an effort to catch the clock up, bits are added to the counter, causing the clock to update quicker and catch up to the correct time. The advantage of this method is that it does not introduce any disparity in the system time, that is noticeable to an end user, or that could corrupt call timeouts and timestamps.

10.9 IP QoS via Differentiated Services (DiffServ)

DiffServ is architecture providing different types or levels of service for IP traffic. DiffServ (according to RFC 2474) offers the capability to prioritize certain traffic types, depending on their priority, thereby accomplishing a higher-level QoS at the expense of other traffic types. By prioritizing packets, DiffServ routers can minimize transmission delays for time-sensitive packets such as VoIP packets.

The Mediant 1000 can be configured to set a different DiffServ value to IP packets according to their class-of-service (Network, Premium Media, Premium Control, Gold and Bronze).

For the mapping of an application to its class-of-service, refer to Table 10-1 on page 310.

The DiffServ parameters are described in Table 5-23 on page 139.

10.10 VLANS and Multiple IPs

10.10.1 Multiple IPs

Media, Control, and Management (OAM) traffic in the gateway can be assigned one of the following IP addressing schemes:

- Single IP address for all traffic (i.e., Media, Control, and OAM).
- Separate IP address for each traffic type.

  For separate IP addresses, the different traffic types are separated into three dedicated networks. Instead of a single IP address, the gateway is assigned three IP addresses and subnet masks, each relating to a different traffic type. This architecture enables users to integrate the gateway into a three-network environment that is focused on security and segregation. Each entity in the gateway (e.g., Web and RTP) is mapped to a single traffic type (according to Table 10-1 on page 310) in which it operates.
- Two separate IP addresses (Dual IP mode)--one for a specific traffic type and the other for a combination of two traffic types.

  In Dual IP mode, the gateway is assigned two IP addresses for the different traffic types. One IP address is assigned to a combination of two traffic types (Media and Control, OAM and Control, or OAM and Media), while the other IP address is assigned to whichever traffic type that is not included in this combination. For example, a typical scenario using this mode would include one IP address assigned for Control and OAM, and another IP address assigned for Media.

Notes:

- A default gateway is supported only for the Media traffic type; for the other two, use the IP Routing table.
- The IP address and subnet mask used in the Single IP Network mode are carried over to the OAM traffic type in the Multiple IP Network mode.

For detailed information on integrating the Mediant 1000 into a VLAN and multiple IPs network, refer to Section 10.10.3 on page 311. For detailed information on configuring the multiple IP parameters, refer to Section 5.6.1.1 on page 139.

10.10.2 IEEE 802.1p/Q (VLANs and Priority)

The Virtual Local Area Network (VLAN) mechanism enables the Mediant 1000 to be integrated into a VLAN-aware environment that includes switches, routers and endpoints.

When in VLAN-enabled mode, each packet is tagged with values that specify its priority (class-of-service) (IEEE 802.1p) and the identifier (traffic type) of the VLAN to which it belongs (media, control or management) (IEEE 802.1Q).

The class-of-service mechanism can be utilized to accomplish Ethernet QoS. Packets sent by the Mediant 1000 to the Ethernet network are divided into five, different-priority classes (Network, Premium media, Premium control, Gold and Bronze). The priority of each class is determined by a corresponding *ini* file parameter.

Traffic type tagging can be used to implement Layer 2 VLAN security. By discriminating traffic into separate and independent domains, the information is preserved within the VLAN. Incoming packets received from an incorrect VLAN are discarded.

For the mapping of an application to its class-of-service and traffic type, refer to Table 10-1 below.

Media traffic type is assigned 'Premium media' class of service, Management traffic type is assigned 'Bronze' class of service, and Control traffic type is assigned 'Premium control' class of service.

For example, RTP/RTCP traffic is assigned the Media VLAN ID and 'Premium media' class of service, whereas Web traffic is assigned the Management VLAN ID and 'Bronze' class of service. Each of these parameters can be configured with an 802.1p/q value: traffic type to VLAN ID, and class of service to 802.1p priority.

Notes:

- As a safety measure, the VLAN mechanism is activated only when the gateway is loaded from the flash memory. Therefore, when using BootP:
  Load an ini file with 'VlanMode = 1' and 'SaveConfiguration = 1'. Then (after the gateway is active) reset the gateway with TFTP disabled, or by using any method except for BootP.
- The gateway must be connected to a VLAN-aware switch, and the switch's PVID must be equal to the gateway's native VLAN ID.

For information on how to configure VLAN parameters, refer to Section 5.6.1.5 on page 147.

Table 10-1: Traffic / Network Types and Priority (continues on pages 310 to 310)

Application	Traffic / Network Types	Class-of-Service (Priority)
Debugging interface	Management	Bronze
Telnet	Management	Bronze
DHCP	Management	Network
Web server (HTTP)	Management	Bronze
SNMP GET/SET	Management	Bronze
Web server (HTTPS)	Management	Bronze
IPSec IKE	Determined by the service	Determined by the service
RTP traffic	Media	Premium media
RTCP traffic	Media	Premium media
T.38 traffic	Media	Premium media
SIP	Control	Premium control
SIP over TLS (SIPS)	Control	Premium control
Syslog	Management	Bronze
ICMP	Management	Determined by the initiator of the request
ARP listener	Determined by the initiator of the request	Network
SNMP Traps	Management	Bronze
DNS client	EnableDNSasOAM	Network
NTP	EnableNTPasOAM	Depends on the traffic type: ▪ Control: Premium control ▪ Management: Bronze
NFS	NFSServers_VlanType in the NFSServers table	Gold

10.10.2.1 Operation

- **Outgoing packets (from the gateway to the switch):** All outgoing packets are tagged, each according to its interface (control, media or OAM). If the gateway's native ID is identical to one of the other IDs (usually to the OAM ID), this ID (e.g., OAM) is set to zero on outgoing packets (VlanSendNonTaggedOnNative = 0). This method is called Priority Tagging (p tag without Q tag). If the parameter VlanSendNonTaggedOnNative is set to 1, the gateway sends regular packets (with no VLAN tag).
- **Incoming packets (from the switch to the gateway):** The switch sends all packets intended for the gateway (according to the switch's configuration) to the gateway without altering them. For packets whose VLAN ID is identical to the switch's PVID. In this case, the switch removes the tag and sends a packet.

The gateway only accepts packets that have a VLAN ID identical to one of its interfaces (control, media or OAM). Packets with a VLAN ID that is 0 or packets without a tag are accepted only if the gateway's native VLAN ID is identical to the VLAN ID of one of its interfaces. In this case, the packets are sent to the relevant interface. All other packets are rejected.

10.10.3 Getting Started with VLANS and Multiple IPs

By default the Mediant 1000 operates without VLANs and multiple IPs, using a single IP address, subnet mask and default gateway IP address. This section provides an example of the configuration required to integrate the Mediant 1000 into a VLAN and multiple IPs network using the Embedded Web Server (refer to Section 10.10.3.1) and *ini* file (refer to Section 10.10.3.2 on page 314). Table 10-2 below shows an example configuration that is implemented in the following sections.

Table 10-2: Example of VLAN and Multiple IPs Configuration

Network Type	IP Address	Subnet Mask	Default Gateway IP Address	VLAN ID	External Routing Rule
OAM	10.31.174.50	255.255.0.0	0.0.0.0	4	83.4.87.X
Control	10.32.174.50	255.255.0.0	0.0.0.0	5	130.33.4.6
Media	10.33.174.50	255.255.0.0	10.33.0.1	6	--

Note that since a default gateway is available only for the Media network, for the Mediant 1000 to be able to communicate with an external device / network on its OAM and Control networks, IP routing rules must be used.

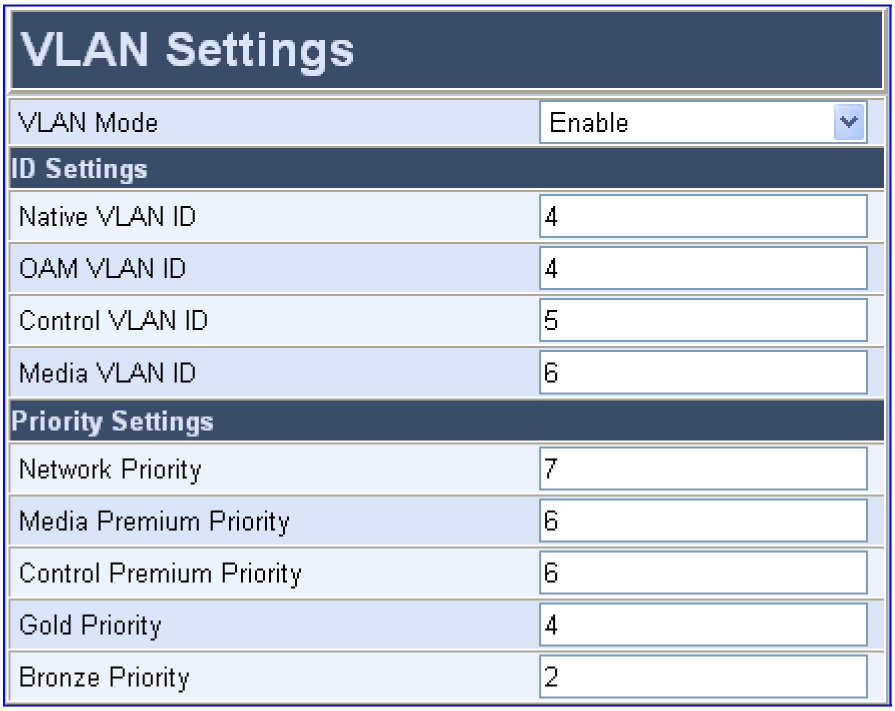
Note: The values provided in Sections 10.10.3.1 and 10.10.3.2 are sample parameter values only and are to be replaced with actual values appropriate to your system.

10.10.3.1 Integrating Using the Embedded Web Server

➢ **To integrate the Mediant 1000 into a VLAN and multiple IPs network using the Embedded Web Server, take these 7 steps:**

1. Access the Embedded Web Server (Section 5.3 on page 57).
2. Use the Software Upgrade Wizard (Section 5.8.1 on page 191) to load and *burn* the firmware version to the Mediant 1000 (VLANs and multiple IPs support is available only when the firmware is burned to flash).
3. Configure the VLAN parameters by completing the following steps:
   - Open the 'VLAN Settings' screen (**Advanced Configuration** menu > **Network Settings** > **VLAN Settings** option); the 'VLAN Settings' screen is displayed.
   - Modify the VLAN parameters to correspond to the values shown in Figure 10-2.

Figure 10-2: Example of the VLAN Settings Screen



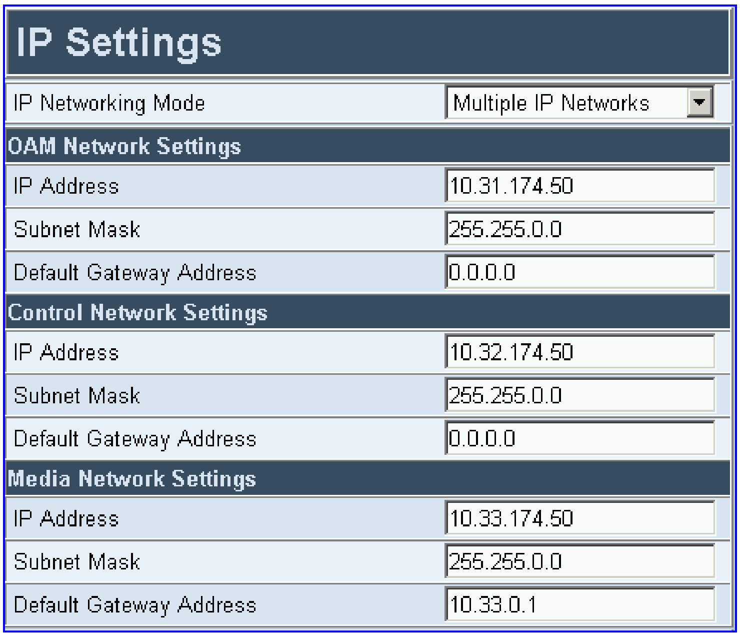
VLAN Settings

VLAN Mode	Enable
ID Settings	
Native VLAN ID	4
OAM VLAN ID	4
Control VLAN ID	5
Media VLAN ID	6
Priority Settings	
Network Priority	7
Media Premium Priority	6
Control Premium Priority	6
Gold Priority	4
Bronze Priority	2

   - Click the **Submit** button to save your changes.
4. Configure the multiple IP parameters by completing the following steps:
   - Open the 'IP Settings' screen (**Advanced Configuration** menu > **Network Settings** > **IP Settings** option); the 'IP Settings' screen is displayed.
   - Modify the IP parameters to correspond to the values shown in Figure 10-3. Note that the OAM, Control and Media Network Settings parameters appear only after you select the options 'Multiple IP Networks' or 'Dual IP' in the field 'IP Networking Mode'.

Note: Configure the OAM parameters only if the OAM networking parameters are different from the networking parameters used in the Single IP Network mode.

Figure 10-3: Example of the IP Settings Screen



IP Settings

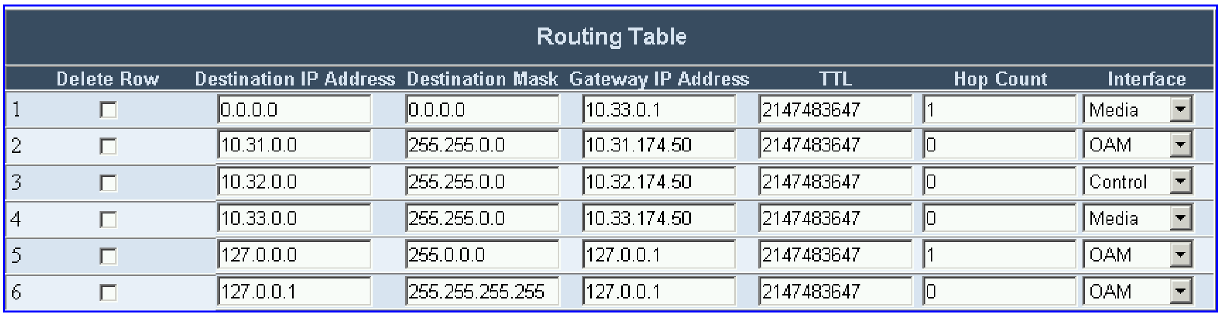
IP Networking Mode	Multiple IP Networks
OAM Network Settings	
IP Address	10.31.174.50
Subnet Mask	255.255.0.0
Default Gateway Address	0.0.0.0
Control Network Settings	
IP Address	10.32.174.50
Subnet Mask	255.255.0.0
Default Gateway Address	0.0.0.0
Media Network Settings	
IP Address	10.33.174.50
Subnet Mask	255.255.0.0
Default Gateway Address	10.33.0.1

- Click the **Submit** button to save your changes.

5. Configure the IP Routing table by completing the following steps (the IP Routing table is required to define static routing rules for the OAM and Control networks since a default gateway isn't supported for these networks):

   - Open the 'IP Routing Table' screen (**Advanced Configuration** menu > **Network Settings** > **IP Routing Table** option); the 'IP Routing Table' screen is displayed.

Figure 10-4: Example of the IP Routing Table Screen



Routing Table

	Delete Row	Destination IP Address	Destination Mask	Gateway IP Address	TTL	Hop Count	Interface
1	☐	0.0.0.0	0.0.0.0	10.33.0.1	2147483647	1	Media
2	☐	10.31.0.0	255.255.0.0	10.31.174.50	2147483647	0	OAM
3	☐	10.32.0.0	255.255.0.0	10.32.174.50	2147483647	0	Control
4	☐	10.33.0.0	255.255.0.0	10.33.174.50	2147483647	0	Media
5	☐	127.0.0.0	255.0.0.0	127.0.0.1	2147483647	1	OAM
6	☐	127.0.0.1	255.255.255.255	127.0.0.1	2147483647	0	OAM

   - Use the 'Add a new table entry' pane to add the routing rules shown in Table 10-3 below.

Table 10-3: Example of IP Routing Table Configuration

Destination IP Address	Destination Mask	Gateway IP Address	Hop Count	Network Type
130.33.4.6	255.255.255.255	10.32.0.1	20	Control
83.4.87.6	255.255.255.0	10.31.0.1	20	OAM

   - Click the **Submit** button to save your changes.

6. Save your changes to flash so they are available after a power fail, refer to Section 5.9.2 on page 200.

7. Reset the gateway (refer to Section 5.9.3 on page 200).

10.10.3.2 Integrating using the *ini* File

➢ To integrate the Mediant 1000 into a VLAN and multiple IPs network using the *ini* file, take these 3 steps:

1. Prepare an *ini* file with parameters shown in Figure 10-5 (refer to the following notes):
   - If the BootP/TFTP utility and the OAM interface are located in the same network, the Native VLAN ID (VlanNativeVlanId) must be equal to the OAM VLAN ID (VlanOamVlanId), which in turn must be equal to the PVID of the switch port the gateway is connected to. Therefore, set the PVID of the switch port to 4 (in this example).
   - Configure the OAM parameters (LocalOAMPAddress, LocalOAMSubnetMask and LocalOAMDefaultGW) only if the OAM networking parameters are different from the networking parameters used in the Single IP Network mode.
   - The IP Routing table is required to define static routing rules for the OAM and Control networks since a default gateway isn't supported for these networks.

Figure 10-5: Example of VLAN and Multiple IPs *ini* File Parameters

```
; VLAN Configuration
VlanMode=1
VlanOamVlanId=4
VlanNativeVlanId=4
VlanControlVlanId=5
VlanMediaVlanID=6

; Multiple IPs Configuration
EnableMultipleIPs=1
LocalMediaIPAddress=10.33.174.50
LocalMediaSubnetMask=255.255.0.0
LocalMediaDefaultGW=10.33.0.1
LocalControlIPAddress=10.32.174.50
LocalControlSubnetMask=255.255.0.0
LocalControlDefaultGW=0.0.0.0
LocalOAMPAddress=10.31.174.50
LocalOAMSubnetMask=255.255.0.0
LocalOAMDefaultGW=0.0.0.0

; IP Routing table parameters
RoutingTableDestinationsColumn = 130.33.4.6, 83.4.87.6
RoutingTableDestinationMasksColumn = 255.255.255.255 , 255.255.255.0
RoutingTableGatewaysColumn = 10.32.0.1 , 10.31.0.1
RoutingTableInterfacesColumn = 1 , 0
RoutingTableHopsCountColumn = 20,20
```

2. Use the BootP/TFTP utility (Section D.6 on page 414) to load and *burn* (-fb option) the firmware version and the *ini* file you prepared in the previous step to the Mediant 1000 (VLANs and multiple IPs support is available only when the firmware is burned to flash).
3. Reset the Mediant 1000 after disabling it on the BootP/TFTP utility.

11 Advanced PSTN Configuration

11.1 Mediant 1000 Clock Settings

The gateway can either generate its own timing signals, using an internal clock, or recover them from one of the E1/T1 trunks.

- **To use the internal gateway clock source configure the following parameters:**
  - TDMBusClockSource = 1
  - ClockMaster = 1 (for all gateway trunks)
- **To use the recovered clock option, configure the following parameters:**
  - TDMBusClockSource = 4
  - ClockMaster_x = 0 (for all ‘slave’ gateway trunks connected to PBX#1)
  - ClockMaster_x = 1 (for all ‘master’ gateway trunks connected to PBX#2)

Assuming that the gateway recovers its internal clock from one of the ‘slave’ trunks connected to PBX#1, and provides clock to PBX#2 on its ‘master’ trunks.

In addition it is necessary to define from which of the ‘slave’ trunks the gateway recovers its clock:

- TDMBusPSTNAutoClockEnable = 1 (the gateway automatically selects one of the connected ‘slave’ trunks)

Or

- TDMBusLocalReference = # (Trunk index: 0 to 3, default = 0)

Note 1: To configure the TDM Bus Clock Source parameters, refer to Section 5.6.4 on page 166.

Note 2: When the Mediant 1000 is used in a ‘non-span’ configuration, the internal gateway clock must be used (as explained above).

11.2 ISDN Overlap Dialing

Overlap dialing is a dialing scheme used by several ISDN variants to send and / or receive called number digits one right after the other (or several at a time). As opposed to en-bloc dialing in which a complete number is sent.

The Mediant 1000 can optionally support ISDN overlap dialing for incoming ISDN calls for the entire gateway by setting ‘ISDNRxOverlap’ to 1, or per E1/T1 span by setting ‘ISDNRxOverlap_x’ to 1 (‘x’ represents the number of the trunk, 0 to 3).

To play a Dial tone to the ISDN user side when an empty called number is received, set ‘ISDNINCallsBehavior = 65536’ (bit #16) causing the Progress Indicator to be included in the SetupAck ISDN message.

The Mediant 1000 stops collecting digits (for ISDN→IP calls) when:

- The sending device transmits a ‘sending complete’ IE in the ISDN Setup or the following INFO messages to signal that no more digits are going to be sent.
- The inter-digit timeout (configured by the parameter ‘TimeBetweenDigits’) expires. The default for this timeout is 4 seconds.
- The maximum allowed number of digits (configured by the parameter ‘MaxDigits’) is reached. The default is 30 digits.
- A match is found with the defined digit map (configured by the parameter, DigitMapping).

Relevant parameters (described in Table 5-33 on page 162):

- ISDNRxOverlap
- ISDNRxOverlap_x
- TimeBetweenDigits
- MaxDigits
- ISDNInCallsBehavior
- DigitMapping

11.3 Using ISDN NFAS

In regular (non-NFAS) T1 ISDN trunks, a single 64 kbps channel carries signaling for the other 23 B-channels of that particular T1 trunk. This channel is called the D-channel and usually resides on timeslot # 24.

The ISDN Non-Facility Associated Signaling (NFAS) feature enables use of a single D-channel to control multiple PRI interfaces.

With NFAS it is possible to define a group of T1 trunks, called an NFAS group, in which a single D-channel carries ISDN signaling messages for the entire group. The NFAS group's B-channels are used to carry traffic, such as voice or data. The NFAS mechanism also enables definition of a backup D-channel on a different T1 trunk, to be used if the primary D-channel fails.

The NFAS group comprises several T1 trunks. Each T1 trunk is called an 'NFAS member'. The T1 trunk whose D-channel is used for signaling is called the 'Primary NFAS Trunk'. The T1 trunk whose D-channel is used for backup signaling is called the 'Backup NFAS Trunk'. The primary and backup trunks each carry 23 B-channels while all other NFAS trunks each carry 24 B-channels.

The Mediant 1000 supports multiple NFAS groups. Each group should contain different T1 trunks.

The NFAS group is identified by an NFAS GroupID number (possible values are 1, 2, 3 and 4). To assign a number of T1 trunks to the same NFAS group, use the parameter 'NFASGroupNumber_x = groupID'. 'x' stands for the physical trunkID (0 to 3).

The parameter 'DchConfig_x = Trunk_type' is used to define the type of NFAS trunk. Trunk_type is set to 0 for the primary trunk, to 1 for the backup trunk and to 2 for an ordinary NFAS trunk. 'x' stands for the physical trunkID (0 to 3).

For example, to assign the first four Mediant 1000 T1 trunks to NFAS group #1, in which trunk #0 is the primary trunk and trunk #1 is the backup trunk, use the following configuration:

```
NFASGroupNumber_0 = 1
NFASGroupNumber_1 = 1
NFASGroupNumber_2 = 1
NFASGroupNumber_3 = 1
DchConfig_0 = 0            ;Primary T1 trunk
DchConfig_1 = 1            ;Backup T1 trunk
DchConfig_2 = 2            ;24 B-channel NFAS trunk
DchConfig_3 = 2            ;24 B-channel NFAS trunk
```

The NFAS parameters are described in Table 5-33 on page 162.

Note: In the current version the NFAS parameters cannot be configured via the 'Trunk Settings' screen in the Embedded Web Server. Use *ini* file configuration instead.

11.3.1 NFAS Interface ID

Several ISDN switches require an additional configuration parameter per T1 trunk that is called ‘Interface Identifier’. In NFAS T1 trunks the Interface Identifier is sent explicitly in Q.931 Setup / Channel Identification IE for all NFAS trunks, except for the B-channels of the Primary trunk (refer to note 1 below).

The Interface ID can be defined per each member (T1 trunk) of the NFAS group, and must be coordinated with the configuration of the Switch.

The default value of the Interface ID is identical to the number of the physical T1 trunk (0 for the first Mediant 1000 trunk, 1 for the second Mediant 1000 T1 trunk, and so on).

To define an explicit Interface ID for a T1 trunk (that is different from the default), use the following parameters:

- ISDNIBehavior_x = 512 (x identifies the Mediant 1000 physical trunk)
- ISDNNFASInterfaceID_x = ID (x = 0 to 255)

Note 1: Usually the Interface Identifier is included in the Q.931 Setup/Channel Identification IE only on T1 trunks that doesn’t contain the D-channel. Calls initiated on B-channels of the Primary T1 trunk, by default, don’t contain the Interface Identifier. Setting the parameter ‘ISDNIBehavior_x’ to 2048’ forces the inclusion of the Channel Identifier parameter also for the Primary trunk.

Note 2: The parameter ‘ISDNNFASInterfaceID_x = ID’ can define the ‘Interface ID’ for any Primary T1 trunk, even if the T1 trunk is not a part of an NFAS group. However, to include the Interface Identifier in Q.931 Setup/Channel Identification IE configure ‘ISDNIBehavior_x = 2048’ in the *ini* file.

11.3.2 Working with DMS-100 Switches

The DMS-100 switch requires the following NFAS Interface ID definitions:

- InterfaceID #0 for the Primary trunk
- InterfaceID #1 for the Backup trunk
- InterfaceID #2 for a 24 B-channel T1 trunk
- InterfaceID #3 for a 24 B-channel T1 trunk
- Etc.

For example, if four T1 trunks on a gateway are configured as a single NFAS group, with Primary and Backup T1 trunks, that is used with a DMS-100 switch, the following parameters should be used:

```
NFASGroupNumber_0 = 1
NFASGroupNumber_1 = 1
NFASGroupNumber_2 = 1
NFASGroupNumber_3 = 1
DchConfig_0 = 0    ;Primary T1 trunk
DchConfig_1 = 1    ;Backup T1 trunk
DchConfig_2 = 2    ;B-Channel NFAS trunk
DchConfig_3 = 2    ;B-channel NFAS trunk
```

If there is no NFAS Backup trunk, the following configuration should be used:

```
ISDNNFASInterfaceID_0 = 0
ISDNNFASInterfaceID_1 = 2
ISDNNFASInterfaceID_2 = 3
ISDNNFASInterfaceID_3 = 4
ISDNIBehavior = 512   ;This parameter should be added because of
;ISDNNFASInterfaceID coniguration above
NFASGroupNumber_0 = 1
NFASGroupNumber_1 = 1
NFASGroupNumber_2 = 1
NFASGroupNumber_3 = 1
DchConfig_0 = 0    ;Primary T1 trunk
DchConfig_1 = 2    ;B-Channel NFAS trunk
DchConfig_2 = 2    ;B-Channel NFAS trunk
DchConfig_3 = 2    ;B-channel NFAS trunk
```

11.4 Redirect Number and Calling Name (Display)

The following tables define the Mediant 1000 redirect number and calling name (Display) support for various PRI variants:

Table 11-1: Calling Name (Display)

	DMS-100	NI-2	4/5ESS	Euro ISDN
NT→TE	Yes	Yes	No	Yes
TE→NT	Yes	Yes	No	No

Table 11-2: Redirect Number

	DMS-100	NI-2	4/5ESS	Euro ISDN
NT→TE	Yes	Yes	Yes	Yes
TE→NT	Yes	Yes	Yes	No

12 Advanced System Capabilities

12.1 Restoring Networking Parameters to their Initial State

You can use the ‘Reset’ button to restore the Mediant 1000 networking parameters to their factory default values (described in Table 4-1) and to reset the username and password.

Note that the Mediant 1000 returns to the software version burned in flash. This process also restores the Mediant 1000 parameters to their factory settings. Therefore, you must load your previously backed-up *ini* file, or the default *ini* file (received with the software kit) to set them to their correct values.

➢ To restore the networking parameters of the Mediant 1000 to their initial state, take these 3 steps:

1. Press in the ‘Reset’ button uninterruptedly for a duration of more than six seconds; the gateway is restored to its factory settings (username: ‘Admin’, password: ‘Admin’).
2. Assign the Mediant 1000 IP address (refer to Section 4.2 on page 47).
3. Load your previously backed-up *ini* file, or the default *ini* file (received with the software kit). To load the *ini* file via the Embedded Web Server, refer to Section 5.6.5 on page 168.

12.2 Establishing a Serial Communications Link with the Mediant 1000

Use serial communication software (e.g., HyperTerminal™) to establish a serial communications link with the Mediant via the RS-232 connection. You can use this link to change the networking parameters (Section 4.2.4 on page 50) and to receive error / notification messages.

➢ To establish a serial communications link with the Mediant 1000 via the RS-232 port, take these 2 steps:

1. Connect the RS-232 port to your PC (refer to Section 3.4.1 on page 40).
2. Use a serial communication software (e.g., HyperTerminal™) with the following communications port settings:
   - Baud Rate: 115,200 bps
   - Data bits: 8
   - Parity: None
   - Stop bits: 1
   - Flow control: None

Note that after resetting the gateway, the information, shown in Figure 12-1, appears on the terminal screen. This information can be used to determine possible Mediant initialization problems, such as incorrectly defined (or undefined) local IP address, subnet mask, etc.

Figure 12-1: RS-232 Status and Error Messages

```
MAC address = 00-90-8F-01-00-9E
Local IP address = 10.1.37.6
Subnet mask = 255.255.0.0
Default gateway IP address = 10.1.1.5
TFTP server IP address = 10.1.1.167
Boot file name = ram35136.cmp
INI file name = mp108.ini
Call agent IP address = 10.1.1.18
Log server IP address = 0.0.0.0
Full/Half Duplex state = HALF DUPLEX
Flash Software Burning state = OFF
Serial Debug Mode = OFF
Lan Debug Mode = OFF
BootLoad Version 1.75
Starting TFTP download... Done.
MP108 Version 3.80.00
```

12.3 Automatic Update Mechanism

The Mediant 1000 is capable of automatically updating its *cmp*, *ini* and configuration files. These files can be stored on any standard Web, FTP or NFS server/s and can be loaded periodically to the gateway via HTTP, HTTPS, FTP or NFS. This mechanism can be used even for Customer Premises Equipment (CPE) devices that are installed behind NAT and firewalls.

The Automatic Update mechanism is applied separately to each file. For the detailed list of available files and their corresponding parameters, refer to 6.5.2 on page 220.

Note: The Automatic Update mechanism assumes the external Web server conforms to the HTTP standard. If the Web server ignores the If-Modified-Since header, or doesn't provide the current date and time during the HTTP 200 OK response, the gateway may reset itself repeatedly. To overcome this problem, adjust the update frequency (AutoUpdateFrequency).

Three methods are used to activate the Automatic Update mechanism:

- After the Mediant 1000 starts-up (refer to the Startup process described in Figure 12-3).
- At a configurable time of the day (e.g., 18:00). This option is disabled by default.
- At fixed intervals (e.g., every 60 minutes). This option is disabled by default.

The following *ini* file example can be used to activate the Automatic Update mechanism.

Figure 12-2: Example of an *ini* File Activating the Automatic Update Mechanism

```
# DNS is required for specifying domain names in URLs
DnsPriServerIP = 10.1.1.11

# Load an extra configuration ini file using HTTP
IniFileURL = 'http://webserver.corp.com/AudioCodes/inifile.ini'
# Load Call Progress Tones file using HTTPS
CptFileUrl = 'https://10.31.2.17/usa_tones.dat'

# Update every day at 03:00 AM
AutoUpdatePredefinedTime = '03:00'
# Note: The cmp file isn’t updated since it is disabled by default
(AutoUpdateCmpFile).
```

Refer to the following notes:

- When HTTP or HTTPS are used, the gateway contacts the Web server/s and queries for the requested files. The *ini* file is loaded only if it was modified since the last automatic update. The *cmp* file is loaded only if its version is different from the version stored on the gateway’s non-volatile memory. All other auxiliary files (e.g., CPT) are updated only once. To update a previously-loaded auxiliary file, you must update the parameter containing its URL.
- To load different configurations (*ini* files) for specific gateways, add the string ‘<MAC>’ to the URL. This mnemonic is replaced with the Mediant 1000 hardware MAC address. Resulting in an *ini* file name request that contains the gateway’s MAC address.
- To automatically update the *cmp* file, use the parameter ‘CmpFileURL’ to specify its name and location. As a precaution (to protect the Mediant 1000 from an accidental update), the Automatic Update mechanism doesn’t apply to the *cmp* file by default. Therefore, (to enable it) set the parameter ‘AutoUpdateCmpFile’ to 1.

The following example illustrates how to utilize Automatic Updates for deploying devices with minimum manual configuration.

➢ To utilize Automatic Updates for deploying the Mediant 1000 with minimum manual configuration, take these 5 steps:

1. Set up a Web server (in the following example it is http://www.corp.com/) where all configuration files are to be stored.
2. To each device, pre-configure the following parameter (DHCP / DNS are assumed): IniFileURL = 'http://www.corp.com/master_configuration.ini'
3. Create a file named master_configuration.ini, with the following text:

```
# Common configuration for all devices
# ------------------------------------
CptFileURL = 'http://www.corp.com/call_progress.dat'
# Check for updates every 60 minutes
AutoUpdateFrequency = 60

# Additional configuration per device
# -----------------------------------
# Each device loads a file named after its MAC address,
# (e.g., config_00908F033512.ini)
IniFileURL = 'http://www.corp.com/config_<MAC>.ini'

# Reset the device after configuration is updated.
# The device resets after all of the files are processed.
ResetNow = 1
```

You can modify the master_configuration.ini file (or any of the config_<MAC>.ini files) at any time. The Mediant 1000 queries for the latest version every 60 minutes and applies the new settings immediately.

4. For additional security, use HTTPS or FTPS. The Mediant 1000 supports HTTPS (RFC 2818) and FTPS using the AUTH TLS method <draft-murray-auth-ftp-ssl-16> for the Automatic Update mechanism.
5. To load configuration files from an NFS server, the NFS file system parameters should be defined in the configuration *ini* file. The following is an example of an *ini* file for loading files from NFS servers using NFS version 2.

```
# Define NFS servers for Automatic Update
[ NFSServers ]
FORMAT NFSServers_Index = NFSServers_HostOrIP, NFSServers_RootPath,
NFSServers_NfsVersion;
NFSServers 1 = 10.31.2.10, /usr/share, 2 ;
NFSServers 2 = 192.168.100.7, /d/shared, 2 ;
[ \NFSServers ]

CptFileUrl = 'file://10.31.2.10/usr/share/public/usa_tones.dat'
VpFileUrl = 'file://192.168.100.7/d/shared/audiocodes/voiceprompt.dat'
```

12.4 Startup Process

The startup process (illustrated in Figure 12-3 on page 324) begins when the gateway is reset (physically or from the Web / SNMP) and ends when the operational software is running. In the startup process, the network parameters, software and configuration files are obtained.

After the gateway powers up or after it is physically reset, it broadcasts a BootRequest message to the network. If it receives a reply (from a BootP server), it changes its network parameters (IP address, subnet mask and default gateway address) to the values provided. If there is no reply from a BootP server and if DHCP is enabled (DHCPEnable = 1), the gateway initiates a standard DHCP procedure to configure its network parameters.

After changing the network parameters, the gateway attempts to load the *cmp* and various configuration files from the TFTP server's IP address, received from the BootP/DHCP servers. If a TFTP server's IP address isn't received, the gateway attempts to load the software (*cmp*) file and / or configuration files from a preconfigured TFTP server (refer to Section 12.3 on page 320). Thus, the gateway can obtain its network parameters from BootP or DHCP servers and its software and configuration files from a different TFTP server (preconfigured in *ini* file).

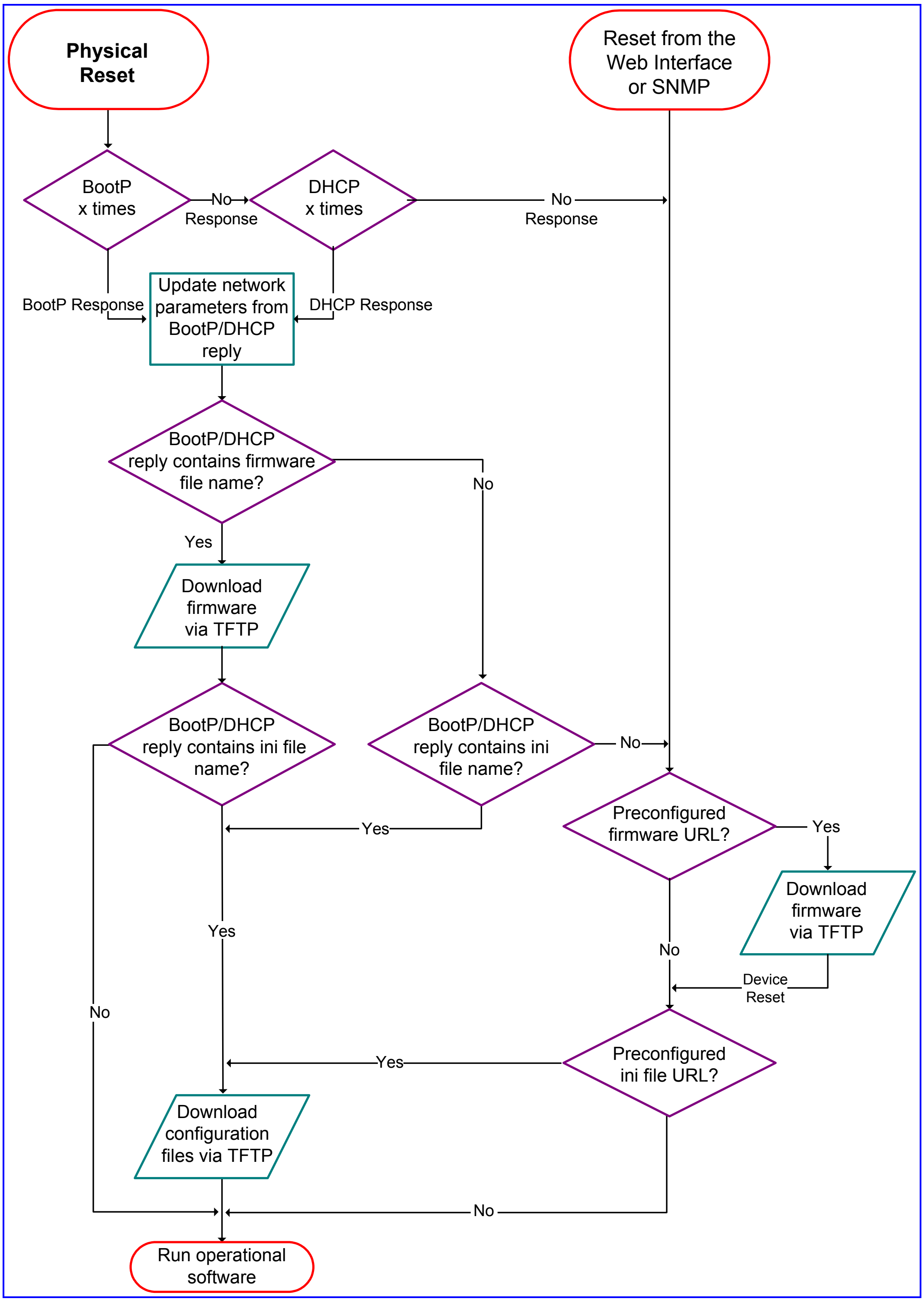
If BootP/DHCP servers are not found or when the gateway is reset from the Web / SNMP, it retains its network parameters and attempts to load the software (*cmp*) file and / or configuration files from a preconfigured TFTP server.

If a preconfigured TFTP server doesn't exist, the gateway operates using the existing software and configuration files loaded on its non-volatile memory.

Notes:

- After the operational software runs, if DHCP is configured, the gateway attempts to renew its lease with the DHCP server.
- Though DHCP and BootP servers are very similar in operation, the DHCP server includes some differences that could prevent its operation with BootP clients. However, many DHCP servers, such as Windows™ NT DHCP server, are backward-compatible with BootP protocol and can be used for gateway configuration.
- The time duration between BootP/DHCP requests is set to 1 second by default. This can be changed by the BootPDelay *ini* file parameter. Also, the number of requests is 3 by default and can be changed by BootPRetries *ini* file parameter (both parameters can also be set using the BootP command line switches).

Figure 12-3: Mediant 1000 Startup Process


Physical Reset
Reset from the Web Interface or SNMP
BootP x times
No Response
DHCP x times
No Response
BootP Response
Update network parameters from BootP/DHCP reply
DHCP Response
BootP/DHCP reply contains firmware file name?
No
Yes
Download firmware via TFTP
BootP/DHCP reply contains ini file name?
BootP/DHCP reply contains ini file name?
No
Yes
Preconfigured firmware URL?
Yes
Download firmware via TFTP
Yes
No
No
Device Reset
Preconfigured ini file URL?
Yes
Download configuration files via TFTP
No
Run operational software

12.5 Using Parameter Tables

The Mediant 1000 uses parameter tables to group related parameters of specific entities and manage them together. These tables, similar to regular parameters, can be configured via the *ini* file, Embedded Web Server, SNMP, etc.

Tables are composed of lines and columns. Columns represent parameters' types. Lines represent specific entities. The instances in each line are called line attributes. Lines in table may represent (for example) a trunk, an NFS file system, list of timers for a given application, etc.

Table 12-1 and Table 12-2 below provide useful examples for reference.

Table 12-1: Example of Parameter Table - Remote Management Connections

Index Fields:
1. Connection Number

Connection Number	User Name	User Password	Time Connected (msec)	Permissions
0	Admin	Yellow9	0	All
1	Gillian	Red5	1266656	Read Only
2	David	Orange6	0	Read Write

Table 12-2: Example of Parameter Table - Port-to-Port Connections

Index Fields:
1. Source Ports
2. Destination IP
3. Destination Port

Source Port	Destination IP	Destination Port	Connection Name	Application Type
2020	10.4.1.50	2020	ATM_TEST_EQ	LAB_EQ
2314	212.199.201.20	4050	ATM_ITROP_LOOP	LAB_EQ
6010	10.3.3.41	6010	REMOTE_MGMT	MGMT

Note: Table 12-1 and Table 12-2 are provided as examples for the purpose of illustration only and are not actually implemented in the Mediant 1000.

12.5.1 Table Indices

Each line in a table must be unique. Therefore, each table defines one or more Index fields. The combination of the Index fields determines the 'line-tag'. Each line-tag appears only once.

In the example provided in Table 12-1 there is only one Index field. This is the simplest way to mark lines.

In the example provided in Table 12-2 there are three Index fields. This more complicated method is a result of the application it represents.

12.5.2 Table Permissions

Each column has a 'permission' attribute that is applied to all instances in the column. This permission determines if and when a field can be modified. Several permissions can be applied to each column.

The following permissions are available:

- **Read:** The value of the field can be read.
- **Write:** The value of the field can be modified.
- **Create:** A value for the field must be provided at creation time (the default values, set to all fields, determine the initial values).
- **Maintenance Write:** The value of the field can only be modified when the entity represented by the line is in maintenance state (each table includes rules that determine when it is in maintenance state).

In the example in Table 12-1 it is assumed that the columns ‘User Name’ and ‘User Password’ have Read-Create permissions. The column ‘Time Connected’ has a Read permission, and the column ‘Permissions’ has Read-Create-Maintenance Write permissions.

12.5.3 Dynamic Tables vs. Static Tables

- **Static Tables:** Static tables don’t support adding new lines or removing (deleting) existing lines. All lines in a Static table are pre-configured with default values. Users can only modify the values of the existing lines. After reset, all lines in a Static table are available.
- **Dynamic Tables:** Dynamic tables support adding and removing lines. They are always initialized as empty tables with no lines. Users should add lines to a Dynamic table via the *ini* file or at run-time via the Embedded Web Server for example.

Note: Certain dynamic tables may initialize a line (or more) at start-up. If so, it is explained in the specific table's documentation.

12.5.4 Secret Tables

A table is defined as a secret table if it contains at least a single secret data field or if it depends on another secret table. A secret data field is a field that mustn’t be revealed to the user. For example, in the IPSec application, IPSec tables are defined as secret tables as the IKE table contains a pre-shared key that must be concealed. Therefore, the SPD table that depends on the IKE table is defined as a secret table as well.

There are two major differences between tables and secret tables:

- The secret field itself cannot be viewed via SNMP, Web or any other application.
- ini file behavior: Secret tables are never displayed in an uploaded *ini* file (e.g., when performing a 'Get *ini* File from Web' operation). Instead, there is a commented title that states that the secret table is present at the gateway and is not to be revealed. Secret tables are always kept in the gateway’s non-volatile memory and can be over-written by new tables that are provided in a new ini file. If a secret table appears in an ini file, it replaces the current table regardless of its content. To delete a secret table from the gateway, provide an empty table of the same type (with no data lines) as part of a new ini file; the empty table replaces the previous table in the gateway.

12.5.5 Using the *ini* File to Configure Parameter Tables

You can use the *ini* file to add / modify parameter tables. When using tables, Read-Only parameters are not loaded, as they cause an error when trying to reload the loaded file. Therefore, Read-Only parameters mustn't be included in tables in the *ini* file. Consequently, tables are loaded with all parameters having at least one of the following permissions: Write, Create or Maintenance Write.

Parameter tables (in an uploaded *ini* file) are grouped according to the applications they configure (e.g., NFS, IPSec). When loading an *ini* file to the gateway, the recommended policy is to include only tables that belong to applications that are to be configured (Dynamic tables of other applications are empty, but static tables are not).

The *ini* file includes a Format line that defines the columns of the table to be modified (this may vary from *ini* file to *ini* file for the same table). The Format line must only include columns that can be modified (parameters that are not specified as Read-Only).

An exception is Index-fields that are always mandatory. In the example provided in Table 12-1, all fields except for the 'Time Connected' field are loaded.

12.5.5.1 Structure of Parameter Tables in the *ini* File

Tables are composed of four elements:

- **The title of the table:** The name of the table in square brackets (e.g., [MY_TABLE_NAME]).
- **A Format line:** Specifies the columns of the table (by their string names) that are to be configured.
  - The first word of the Format line must be 'FORMAT', followed by the names of the Indices fields, and an equal sign '='. After the equal sign the names of the columns are listed.
  - Items must be separated by a comma ','.
  - The Format line must end with a semicolon ';'.
- **Data line(s):** Contain the actual values of the parameters. The values are interpreted according to the Format line. The first word of the Data line must be the table's string name followed by the Index fields.
  - Items must be separated by a comma ','.
  - A Data line must end with a semicolon ';'.
- **End-of-Table-Mark:** Indicates the end of the table. The same string used for the table's title, preceded by a backslash '\' (e.g., [\MY_TABLE_NAME]).

Figure 12-4 displays an example of the structure of a parameter table in the *ini* file.

Figure 12-4: Structure of a Parameter Table in the *ini* File

```
; Table: Items Table.
; Fields: Item_Name, Item_Serial_Number, Item_Color, Item_weight.
; NOTE: Item_Color is not specified. It will be given default value.
[Items_Table]
; Fields declaration
Format Item_Index =  Item_Name, Item_Serial_Number, Item_weight;
Items_Table 0 = Computer, 678678, 6;
Items_Table 6 = Computer-screen, 127979, 9;
Items_Table 2 = Computer-pad, 111111, $$;
[\Items_Table]
```

Refer to the following notes:

- Indices (in both the Format and the Data lines) must appear in the same order determined by the specific table's documentation. The Index field must never be omitted.
- The Format line can include a sub-set of the configurable fields in a table. In this case, all other fields are assigned with the pre-defined default values for each configured line.
- The order of the fields in the Format line isn't significant (as opposed to the Index-fields). The fields in the Data lines are interpreted according to the order specified in the Format line.
- The sign '$$' in a Data line indicates that the user wants to assign the pre-defined default value to it.
- The order of the Data lines is insignificant.
- Data lines must match the Format line, i.e., it must contain exactly the same number of Indices and Data fields and must be in exactly the same order.
- A line in a table is identified by its table-name and Index fields. Each such line may appear only once in the *ini* file.
- Table dependencies:
  Certain tables may depend on other tables. For example, one table may include a field that specifies an entry in another table. This method is used to specify additional attributes of an entity, or to specify that a given entity is part of a larger entity. The tables must appear in the order of their dependency (i.e., if Table X is referred to by Table Y, Table X must appear in the *ini* file before Table Y).

12.6 Customizing the Mediant 1000 Web Interface

Customers incorporating the Mediant 1000 into their portfolios can customize the Web Interface to suit their specific corporate logo and product naming conventions.

Customers can customize the Web Interface's title bar (AudioCodes' title bar is shown in Figure 12-5; a customized title bar is shown in Figure 12-7).

Figure 12-5: User-Customizable Web Interface Title Bar

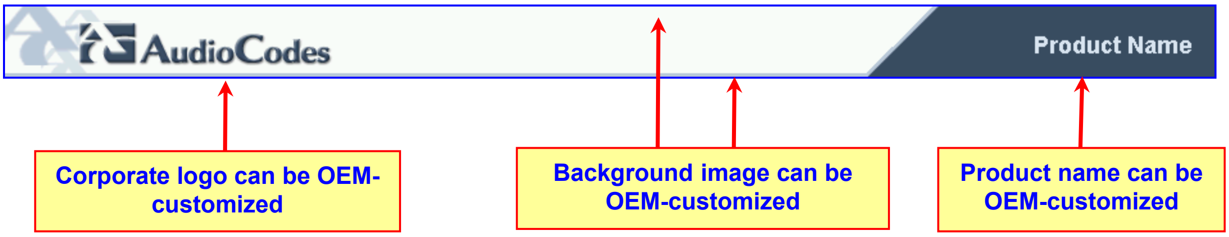
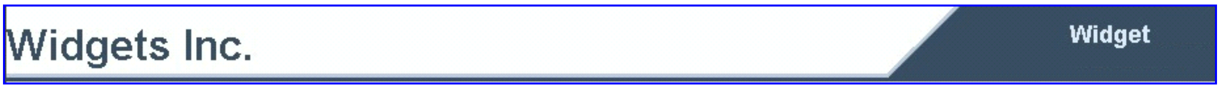

AudioCodes
Product Name
Corporate logo can be OEM-customized
Background image can be OEM-customized
Product name can be OEM-customized

Figure 12-6: Customized Web Interface Title Bar



➢ To customize the title bar via the Web Interface, take these 3 steps:

1. Replace the main corporate logo (refer to Section 12.6.1 below).
2. Replace the title bar's background image file (refer to Section 12.6.2 on page 331).
3. Customize the product's name (refer to Section 12.6.3 on page 332).

12.6.1 Replacing the Main Corporate Logo

The main corporate logo can be replaced either with a different logo image file (refer to Section 12.6.1.1 below) **or** with a text string (refer to Section 12.6.1.2 on page 331). Note that when the main corporation logo is replaced, AudioCodes' logo on the left bar (refer to Figure 5-2) and in the Software Upgrade Wizard (Section 5.8.1 on page 191) disappear.

Also note that the browser's title bar is automatically updated with the string assigned to the WebLogoText parameter when AudioCodes' default logo is not used.

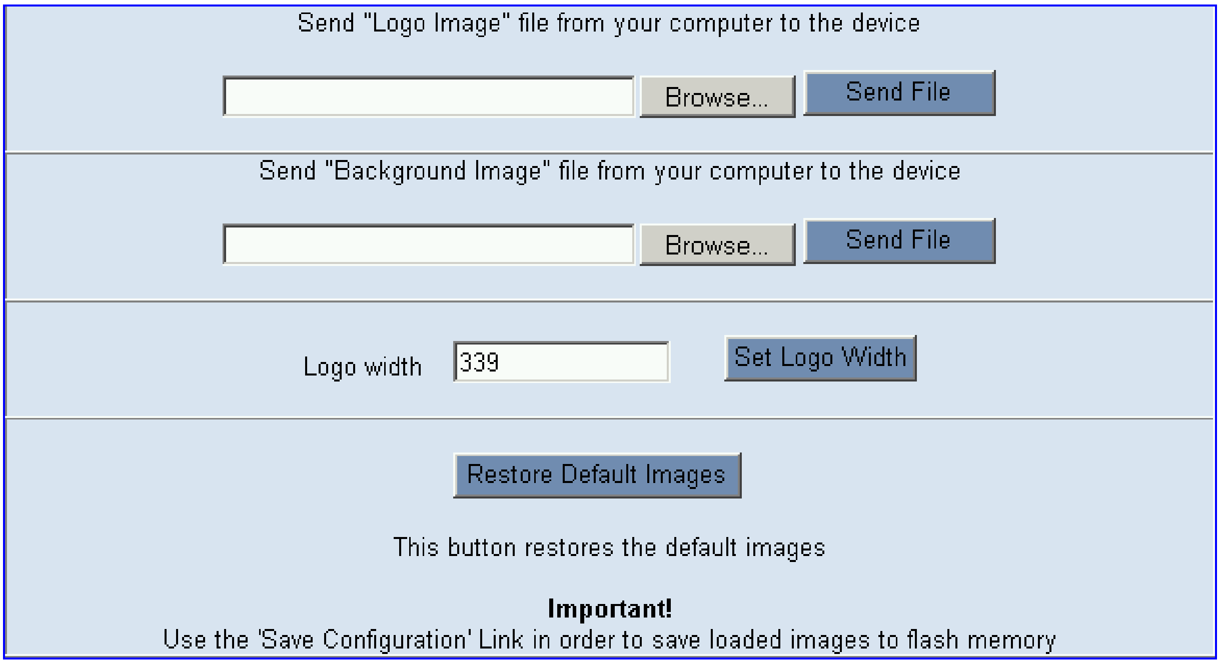
12.6.1.1 Replacing the Main Corporate Logo with an Image File

Note: Use a gif, jpg or jpeg file for the logo image. It is important that the image file has a fixed height of 59 pixels (the width can be configured up to a maximum of 339 pixels). The size of the image files (logo and background) is limited each to 64 kbytes.

➢ **To replace the default logo with your own corporate image via the Web Interface, take these 7 steps:**

1. Access the Mediant 1000 Embedded Web Server (refer to Section 5.3 on page 57).
2. In the URL field, append the suffix 'AdminPage' (note that it's case-sensitive) to the IP address, e.g., http://10.1.229.17/AdminPage.
3. Click **Image Load to Device**; the Image Download screen is displayed (shown in Figure 12-7).

Figure 12-7: Image Download Screen



Send "Logo Image" file from your computer to the device

Send "Background Image" file from your computer to the device

Logo width 339

This button restores the default images

Important!
Use the 'Save Configuration' Link in order to save loaded images to flash memory

4. Click the **Browse** button in the **Send Logo Image File from your computer to the device** box. Navigate to the folder that contains the logo image file you want to load.
5. Click the **Send File** button; the file is sent to the device. When loading is complete, the screen is automatically refreshed and the new logo image is displayed.

6. Note the appearance of the logo. If you want to modify the width of the logo (the default width is 339 pixels), in the 'Logo Width' field, enter the new width (in pixels), and then click the **Set Logo Width** button.
7. To save the image to flash memory so it is available after a power fail, refer to Section 5.9.2 on page 200.

The new logo appears on all Web Interface screens.

Tip: If you encounter any problem during the loading of the files, or you want to restore the default images, click the **Restore Default Images** button.

➢ To replace the default logo with your own corporate image via the *ini* file, take these 2 steps:

1. Place your corporate logo image file in the same folder as where the device's *ini* file is located (i.e., the same location defined in the BootP/TFTP configuration utility). For detailed information on the BootP/TFTP, refer to Appendix D on page 413.
2. Add/modify the two *ini* file parameters in Table 12-3 according to the procedure described in Section 6.2 on page 213.

Note: Loading the device's *ini* file using the 'Configuration File' screen in the Web interface doesn't load the corporate logo image files.

Table 12-3: Customizable Logo *ini* File Parameters

Parameter	Description
LogoFileName	The name of the image file containing your corporate logo. Use a gif, jpg or jpeg image file. The default is AudioCodes' logo file. **Note:** The length of the name of the image file is limited to 47 characters.
LogoWidth	Width (in pixels) of the logo image. **Note:** The optimal setting depends on the resolution settings. The default value is 441 which is the width of AudioCodes' displayed logo.

12.6.1.2 Replacing the Main Corporate Logo with a Text String

The main corporate logo can be replaced with a text string.

- To replace AudioCodes' default logo with a text string *via the Web Interface*, modify the two *ini* file parameters in Table 12-4 according to the procedure described in Section 12.6.4 on page 332.
- To replace AudioCodes' default logo with a text string *via the ini file*, add/modify the two *ini* file parameters in Table 12-4 according to the procedure described in Section 6.2 on page 213.

Table 12-4: Web Appearance Customizable *ini* File Parameters

Parameter	Description
UseWebLogo	0 = Logo image is used (default). 1 = Text string is used instead of a logo image.
WebLogoText	Text string that replaces the logo image. The string can be up to 15 characters.

12.6.2 Replacing the Background Image File

The background image file is duplicated across the width of the screen. The number of times the image is duplicated depends on the width of the background image and screen resolution. When choosing your background image, keep this in mind.

Note: Use a gif, jpg or jpeg file for the background image. It is important that the image file has a fixed height of 59 pixels. The size of the image files (logo and background) is limited each to 64 kbytes.

➢ To replace the background image via the Web, take these 6 steps:

1. Access the Mediant 1000 Embedded Web Server (refer to Section 5.3 on page 57).
2. In the URL field, append the suffix 'AdminPage' (note that it's case-sensitive) to the IP address, e.g., http://10.1.229.17/AdminPage.
3. Click the **Image Load to Device**; the Image load screen is displayed (shown in Figure 12-7).
4. Click the **Browse** button in the 'Send Background Image File from your computer to gateway' box. Navigate to the folder that contains the background image file you want to load.
5. Click the **Send File** button; the file is sent to the device. When loading is complete, the screen is automatically refreshed and the new background image is displayed.
6. To save the image to flash memory so it is available after a power fail, refer to Section 5.9.2 on page 200.

The new background appears on all Web Interface screens.

Tip 1: If you encounter any problem during the loading of the files, or you want to restore the default images, click the **Restore Default Images** button.

Tip 2: When replacing both the background image and the logo image, first load the logo image followed by the background image.

➢ To replace the background image via the *ini* file, take these 2 steps:

1. Place your background image file in the same folder as where the device's *ini* file is located (i.e., the same location defined in the BootP/TFTP configuration utility). For detailed information on the BootP/TFTP, refer to Appendix D on page 413.
2. Add/modify the *ini* file parameters in Table 12-5 according to the procedure described in Section 6.2 on page 213.

> **Note:** Loading the device's *ini* file using the 'Configuration File' screen in the Web interface doesn't load the logo image file.

Table 12-5: Customizable Logo *ini* File Parameters

Parameter	Description
BkgImageFileName	The name (and path) of the file containing the new background. Use a gif, jpg or jpeg image file. The default is AudioCodes background file. **Note:** The length of the name of the image file is limited to 47 characters.

12.6.3 Customizing the Product Name

The Product Name text string can be modified according to OEMs specific requirements.

- To replace AudioCodes' default product name with a text string *via the Web Interface*, modify the two *ini* file parameters in Table 12-6 according to the procedure described in Section 12.6.4 on page 332.
- To replace AudioCodes' default product name with a text string *via the ini file*, add/modify the two *ini* file parameters in Table 12-6 according to the procedure described in Section 6.2 on page 213.

Table 12-6: Web Appearance Customizable *ini* File Parameters

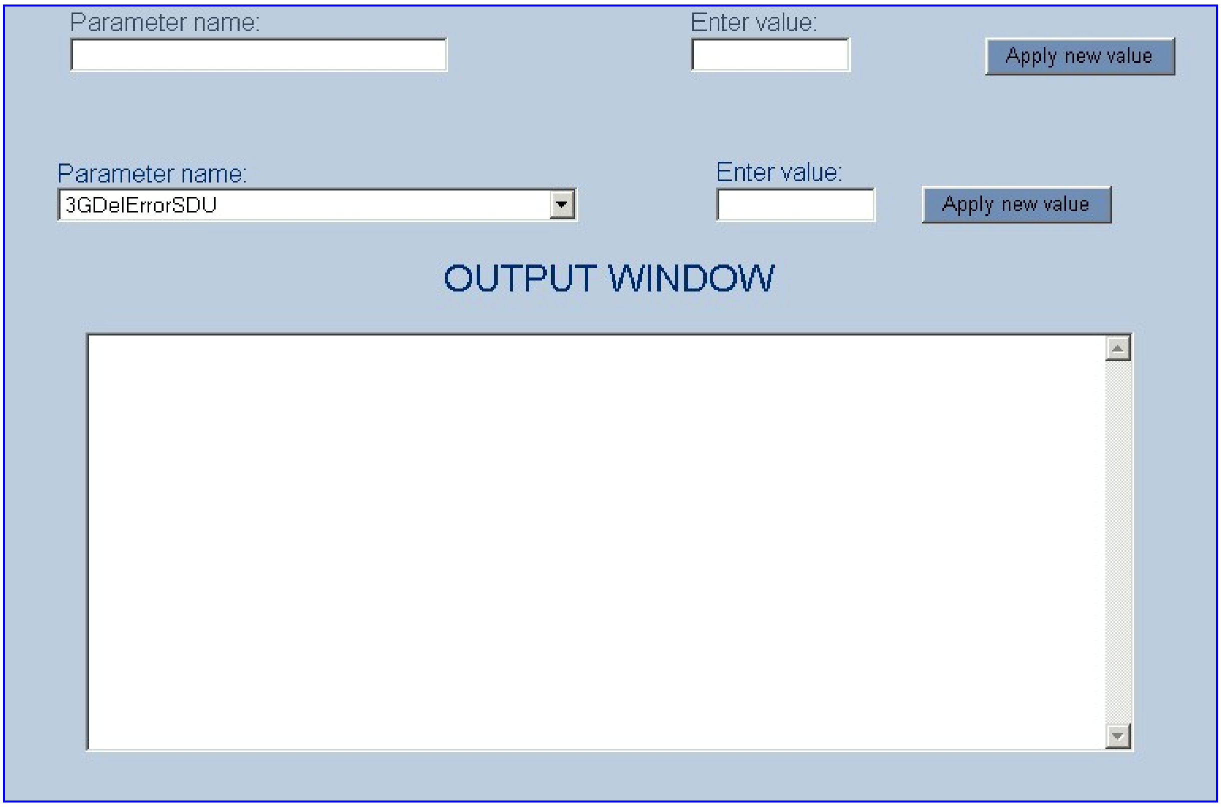
Parameter	Description
UseProductName	0 = Don't change the product name (default). 1 = Enable product name change.
UserProductName	Text string that replaces the product name. The default is 'Mediant 1000'. The string can be up to 29 characters.

12.6.4 Modifying *ini* File Parameters via the Web AdminPage

➢ To modify *ini* file parameters via the AdminPage, take these 6 steps:

1. Access the Mediant 1000 Embedded Web Server (refer to Section 5.3 on page 57).
2. In the URL field, append the suffix 'AdminPage' (note that it's case-sensitive) to the IP address, e.g., http://10.1.229.17/AdminPage.
3. Click the **INI Parameters** option; the INI Parameters screen is displayed (shown in Figure 12-8).

Figure 12-8: INI Parameters Screen



Parameter name:

Enter value:

Parameter name: 3GDelErrorSDU

Enter value:

4. From the 'Parameter Name' drop-down list, select the required *ini* file parameter.
5. In the 'Enter Value' field, enter the parameter's new value.
6. Click the **Apply new value** button to the right; the INI Parameters screen is refreshed, the parameter name with the new value appears in the fields at the top of the screen and the Output Window displays a log displaying information on the operation.

Note: You cannot load the image files (e.g., logo/background image files) to the device by choosing a file name parameter in this screen.

12.7 Software Upgrade Key

Mediant 1000 gateways are supplied with a Software Upgrade Key already pre-configured for the gateway.

Users can later upgrade their Mediant 1000 features, capabilities and quantity of available resources by specifying what upgrades they require, and purchasing a new key to match their specification.

The Software Upgrade Key is sent as a string in a text file, to be loaded into the Mediant 1000. Stored in the device's non-volatile flash memory, the string defines the features and capabilities allowed by the specific key purchased by the user. The device allows users to utilize *only these* features and capabilities. A new key overwrites a previously installed key.

> **Note:** The Software Upgrade Key is an encrypted key. The Software Upgrade Key is provided by AudioCodes only.

12.7.1 Backing up the Current Software Upgrade Key

Back up your current Software Upgrade Key before loading a new key to the device. You can always reload this backed-up key to restore your device capabilities to what they originally were if the 'new' key doesn't comply with your requirements.

➢ To backup the current Software Upgrade Key, take these 5 steps:

1. Access the devices Embedded Web Server (refer to Section 5.3 on page 57).
2. Click the **Software Update** button.
3. Click the **Software Upgrade Key** tab; the Software Upgrade Key screen is displayed (shown in Figure 12-9).
4. Copy the string from the **Current Key** field and paste it in a new file.
5. Save the text file with a name of your choosing.

12.7.2 Loading the Software Upgrade Key

After receiving the Software Upgrade Key file (do not modify its contents in any way), ensure that its first line is [LicenseKeys] and that it contains one or more lines in the following format:

S/N<Serial Number of gateway> = <long Software Upgrade Key>

For example: S/N370604 = jCx6r5tovCIKaBBbhPtT53Yj...

One S/N must match the S/N of your device. The device's S/N can be viewed in the 'Device Information' screen (refer to Section 5.7.3 on page 190).

You can load a Software Upgrade Key using:

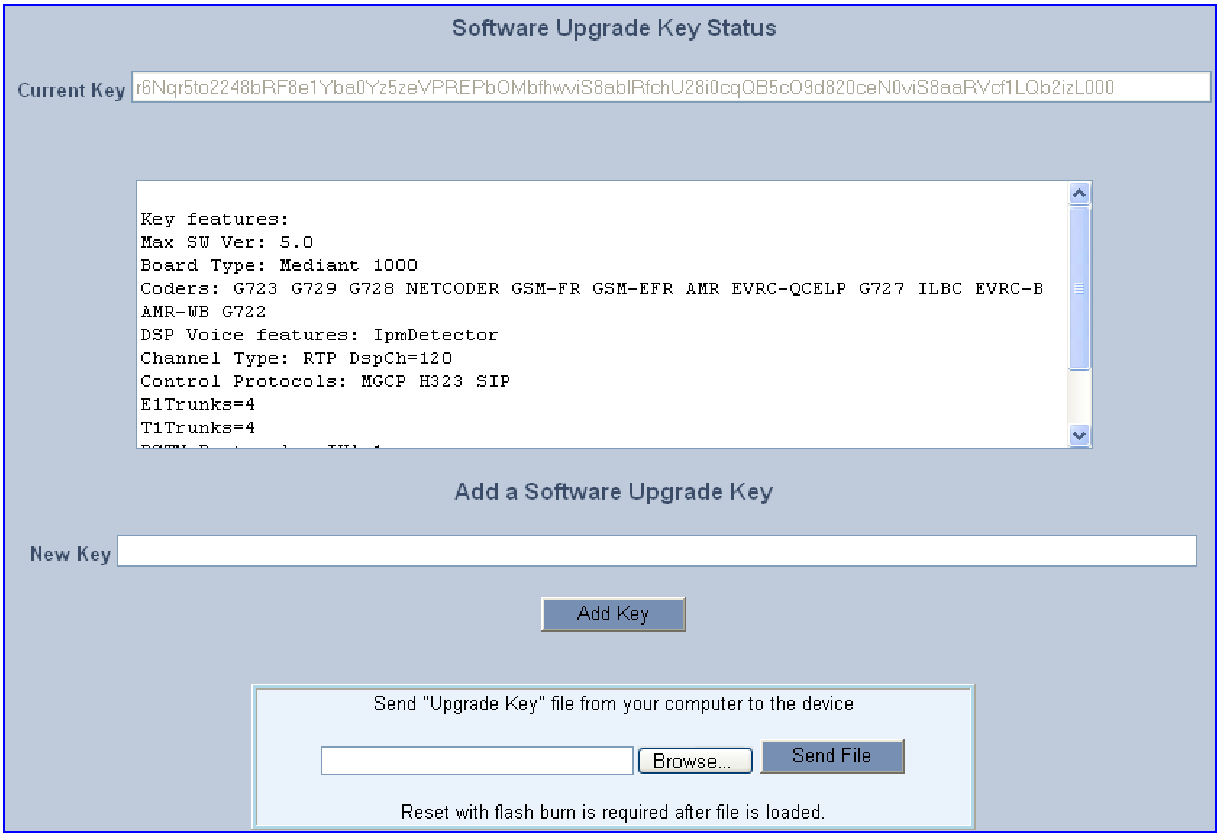
- The Embedded Web Server (refer to Section 12.7.3).
- The BootP/TFTP configuration utility (refer to Section 12.7.4 on page 336).
- AudioCodes' EMS (refer to Section 16.10 on page 388 and to AudioCodes' EMS User's Manual or EMS Product Description).

12.7.3 Loading the Software Upgrade Key Using the Embedded Web Server

➢ **To load a Software Upgrade Key using the Web Server, take these 6 steps:**

1. Access the devices Embedded Web Server (refer to Section 5.3 on page 57).
2. Click the **Software Update** button.
3. Click the **Software Upgrade Key** tab; the Software Upgrade Key screen is displayed (shown in Figure 12-9).

Figure 12-9: Software Upgrade Key Screen



Software Upgrade Key Status

Current Key r6Nqr5to2248bRF8e1Yba0Yz5zeVPREPbOMbfhwviS8ablRfchU28i0cqQB5cO9d820ceN0viS8aaRVcf1LQb2izL000

```
Key features:
Max SW Ver: 5.0
Board Type: Mediant 1000
Coders: G723 G729 G728 NETCODER GSM-FR GSM-EFR AMR EVRC-QCELP G727 ILBC EVRC-B
AMR-WB G722
DSP Voice features: IpmDetector
Channel Type: RTP DspCh=120
Control Protocols: MGCP H323 SIP
E1Trunks=4
T1Trunks=4
```

Add a Software Upgrade Key

New Key

Send "Upgrade Key" file from your computer to the device

Reset with flash burn is required after file is loaded.

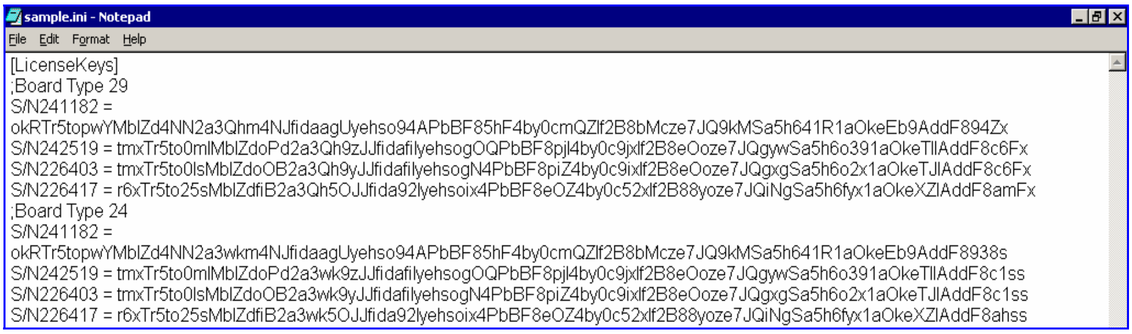
4. **When loading a single key S/N line to a device:**
   a. Open the Software Upgrade Key file (using for example, Notepad).
   b. Copy the key string of the device’s S/N, and then paste it into the field 'New Key'. If the string is sent in the body of an email, copy and paste it from there.
   c. Click the **Add Key** button.

5. **When loading a Software Upgrade Key text file containing multiple S/N lines to a device** (refer to Figure 12-10):
    a. In the 'Send “Upgrade Key” file from your computer to the device' field, click the **Browse** button, and then navigate to the Software Upgrade Key text file.
    b. Click the **Send File** button.

    The new key is loaded to the device, validated, and if valid is burned to memory. The new key is displayed in the 'Current Key' field.

    Validate the new key by scrolling through the ‘Key features’ panel, verifying the presence of the appropriate features.
6. After verifying that the Software Upgrade Key was successfully loaded, reset the device; the new capabilities and resources are active.

Figure 12-10: Example of a Software Upgrade Key File Containing Multiple S/N Lines



```
[LicenseKeys]
;Board Type 29
S/N241182 =
okRTr5topwYMblZd4NN2a3Qhm4NJfidaagUyehso94APbBF85hF4by0cmQZlf2B8bMcze7JQ9kMSa5h641R1aOkeEb9AddF894Zx
S/N242519 = tmxTr5to0mlMblZdoPd2a3Qh9zJJfidafilyehsogOQPbBF8pjl4by0c9jxlf2B8eOoze7JQgywSa5h6o391aOkeTllAddF8c6Fx
S/N226403 = tmxTr5to0lsMblZdoOB2a3Qh9yJJfidafilyehsogN4PbBF8piZ4by0c9ixlf2B8eOoze7JQgxgSa5h6o2x1aOkeTJlAddF8c6Fx
S/N226417 = r6xTr5to25sMblZdfiB2a3Qh5OJJfida92lyehsoix4PbBF8eOZ4by0c52xlf2B88yoze7JQiNgSa5h6fyx1aOkeXZlAddF8amFx
;Board Type 24
S/N241182 =
okRTr5topwYMblZd4NN2a3wkm4NJfidaagUyehso94APbBF85hF4by0cmQZlf2B8bMcze7JQ9kMSa5h641R1aOkeEb9AddF8938s
S/N242519 = tmxTr5to0mlMblZdoPd2a3wk9zJJfidafilyehsogOQPbBF8pjl4by0c9jxlf2B8eOoze7JQgywSa5h6o391aOkeTllAddF8c1ss
S/N226403 = tmxTr5to0lsMblZdoOB2a3wk9yJJfidafilyehsogN4PbBF8piZ4by0c9ixlf2B8eOoze7JQgxgSa5h6o2x1aOkeTJlAddF8c1ss
S/N226417 = r6xTr5to25sMblZdfiB2a3wk5OJJfida92lyehsoix4PbBF8eOZ4by0c52xlf2B88yoze7JQiNgSa5h6fyx1aOkeXZlAddF8ahss
```

12.7.4 Loading the Software Upgrade Key using BootP/TFTP

➢ To load the Software Upgrade Key file using BootP/TFTP, take these 5 steps:

1. Place the file in the same location you’ve saved the *device’s cmp* file. Note that in order to load the Software Upgrade Key via TFTP server, the extension of the key file must be *ini*.
2. Start the BootP/TFTP configuration utility and from the **Services** menu in the main screen, choose option **Clients**; the Client Configuration screen is displayed (refer to Figure D-4 on page 419).
3. From the drop-down list in the **INI File** field, select the Software Upgrade Key file instead of the device’s *ini* file. Note that the device’s *cmp* file must be specified in the **Boot File** field.
4. Configure the initial BootP/TFTP parameters required, and click **OK** (refer to Appendix D on page 413).
5. Reset the device; the device’s *cmp* and Software Upgrade Key files are loaded to the device.

12.7.5 Verifying that the Key was Successfully Loaded

After installing the key, you can determine in the Embedded Web Server’s read-only ‘Key features:’ panel (**Software Update** menu > **Software Upgrade Key**) (refer to Figure 12-9) that the features and capabilities activated by the installed string match those that were ordered.

You can also verify that the key was successfully loaded to the device by accessing the Syslog server. For detailed information on the Syslog server, refer to Section 15.3 on page 366. When a key is successfully loaded, the following message is issued in the Syslog server:

‘S/N___ Key Was Updated. The Board Needs to be Reloaded with *ini* file\n’

12.7.6 Troubleshooting an Unsuccessful Loading of a Key

If the Syslog server indicates that a Software Upgrade Key file was unsuccessfully loaded (the SN_ line is blank), take the following preliminary actions to troubleshoot the issue:

- Open the Software Upgrade Key file and check that the S/N line of the specific device whose key you want to update is listed in it. If it isn’t, contact AudioCodes.
- Verify that you’ve loaded the correct file and that you haven’t loaded the device’s *ini* file or the CPT *ini* file by mistake. Open the file and ensure that the first line is [LicenseKeys].
- Verify that you didn’t alter in any way the contents of the file.

12.7.7 Abort Procedure

Reload the key you backed-up in Section 12.7.1 on page 334 to restore your device capabilities to what they originally. To load the backed-up key use the procedure described in Section 12.7.2 on page 334.

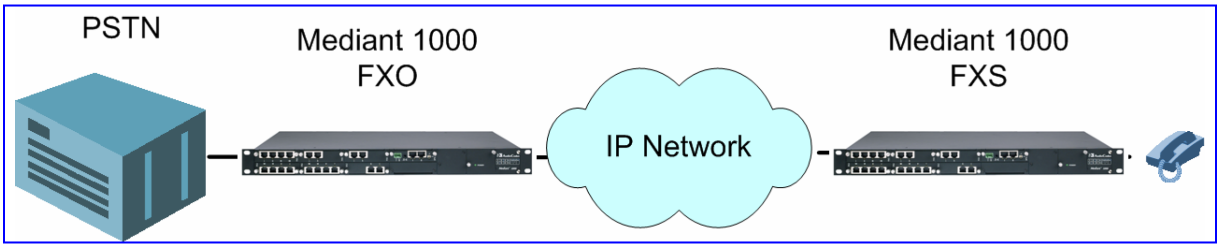
Reader's Notes

13 Special Applications

13.1 Metering Tones Relay

The Mediant 1000 FXS and FXO modules can be used to relay standard 12 or 16 kHz metering tones over the IP network as illustrated in Figure 13-1 below.

Figure 13-1: Metering Tone Relay Architecture



After a call is established between the FXS and FXO modules, the PSTN generates 12 or 16 kHz metering tones towards the FXO module. The FXO module detects these pulses and relays them, over IP, to the FXS module using a proprietary INFO messages (shown in Figure 13-2). The FXS module generates the same pulses to the connected phone.

The parameter ‘MeteringType’ (described in Table 5-9 on page 95) is used to determine the frequency of the metering tone (12 kHz (default) or 16 kHz). In addition, the correct (12 or 16 kHz) coefficient must be used for both FXS and FXO modules.

To enable this feature configure ‘SendMetering2IP = 1’.

The proprietary INFO message used to relay the metering tone pulse contains a ‘Content-Type: message/Metering’:

Figure 13-2: Proprietary INFO Message for Relaying Metering Tones

```
INFO sip:108@10.13.83.1 SIP/2.0
Via: SIP/2.0/UDP 10.13.83.2;branch=z9hG4bKacEizRjAa
Max-Forwards: 70
From: "aviad" <sip:201@10.13.83.2>;tag=1c1638621413
To: <sip:108@10.13.83.1;user=phone>;tag=1c1412617336
Call-ID: 2031013892fcCd@10.13.83.2
CSeq: 3 INFO
Contact: <sip:201@10.13.83.2>
Supported: em,timer,replaces,path
Allow:
REGISTER,OPTIONS,INVITE,ACK,CANCEL,BYE,NOTIFY,PRACK,REFER,INFO,SUBSCRIBE,UPDATE
User-Agent: Audiocodes-Sip-Gateway-Mediant 1000/v.4.40.0.18700
Content-Type: message/Metering
Content-Length: 0
```

13.2 TDM Tunneling

The Mediant 1000 TDM Tunneling feature allows you to tunnel groups of digital trunk spans or timeslots (B-channels) over the IP network. TDM Tunneling utilizes the internal routing capabilities of the Mediant 1000 (working without Proxy control) to receive voice and data streams from TDM (1 to 16 E1/T1/J1) spans or individual timeslots, convert them into packets and transmit them automatically over the IP network (using point-to-point or point-to-multipoint gateway distributions). A Mediant 1000 opposite it (or several Mediant 1000 gateways, when point-to-multipoint distribution is used) converts the IP packets back into TDM traffic. Each timeslot can be targeted to any other timeslot within a trunk in the opposite Mediant 1000.

13.2.1 Implementation

When TDM Tunneling is enabled (‘EnableTDMOverIP’ is set to 1 on the originating Mediant 1000), the originating Mediant 1000 automatically initiates SIP calls from all enabled B-channels belonging to the E1/T1/J1 spans that are configured with the ‘Transparent’ protocol (for ISDN trunks), or ‘Ras CAS’ (for CAS trunks). The called number of each call is the internal phone number of the B-channel that the call originates from. The IP to Trunk Group routing table is used to define the destination IP address of the terminating Mediant 1000. The terminating Mediant 1000 gateway automatically answers these calls if its E1/T1 protocol is set to ‘Transparent’ (ProtocolType = 5), or ‘Raw CAS’ (ProtocolType = 3 for T1 and 9 for E1) and the parameter ‘ChannelSelectMode = 0’ (By Phone Number).

Note: It is possible to configure both gateways to also operate in symmetric mode. To do so, set ‘EnableTDMOverIP’ to 1 and configure the Tel to IP Routing tables in both Mediant 1000 gateways. In this mode, each gateway (after it is reset) initiates calls to the second gateway. The first call for each B-channel is answered by the second gateway.

The Mediant 1000 monitors the established connections continuously, if for some reason one or more calls are released, the gateway automatically reestablishes these ‘broken’ connections. In addition, when a failure in a physical trunk or in the IP network occurs, the Mediant 1000 gateways reestablish the tunneling connections as soon as the network restores.

Note: It is recommended to use the keep-alive mechanism for each connection by activating ‘session expires’ timeout, and using Reinvite messages.

By utilizing the ‘Profiles’ mechanism (refer to Section 5.5.5 on page 115) you can configure the TDM Tunneling feature to choose different settings, based on a timeslot or groups of timeslots. For example, you can use low-bit-rate vocoders to transport voice, and ‘Transparent’ coder to transport data (e.g., for D-channel). You can also use Profiles to assign ToS (for DiffServ) per source, a time-slot carrying data or signaling gets a higher priority value than a time-slot carrying voice.

For tunneling of E1/T1 CAS trunks set the protocol type to Raw CAS (ProtocolType = 3 / 9) and enable RFC 2833 CAS relay mode (CASTransportType = 1).

Figure 13-3 and Figure 13-4 show an example of *ini* files for two Mediant 1000 gateways implementing TDM Tunneling for four E1 spans. Note that in this example both gateways are dedicated to TDM tunneling.

Figure 13-3: *ini* File Example for TDM Tunneling (Originating Side)

```
EnableTDMOverIP = 1

;E1_TRANSPARENT_31
ProtocolType_0 = 5
ProtocolType_1 = 5
ProtocolType_2 = 5
ProtocolType_3 = 5

prefix = '*,10.8.24.12' ;(IP address of the Mediant 1000 in the opposite
location)
; Channel selection by Phone number
ChannelSelectMode = 0
;Profiles can be used do define different coders per B-channels, such as
Transparent
; coder for B-channels (time slot 16) that carries PRI signaling.
TrunkGroup = 0/1-31,1000,1
TrunkGroup = 1/1-31,2000,1
TrunkGroup = 2/1-31,3000,1
TrunkGroup = 3/1-31,4000,1
TrunkGroup = 0/16-16,7000,2
TrunkGroup = 1/16-16,7001,2
TrunkGroup = 2/16-16,7002,2
TrunkGroup = 3/16-16,7003,2

CoderName = 'g7231'
CoderName = 'Transparent'
CoderName_1 = 'g7231'
CoderName_2 = 'Transparent'

TelProfile_1 = voice,$$,1,$$,$$,$$,$$,$$,$$,$$
TelProfile_2 = data,$$,2,$$,$$,$$,$$,$$,$$,$$
```

Figure 13-4: *ini* File Example for TDM Tunneling (Terminating Side)

```
;E1_TRANSPARENT_31
ProtocolType_0 = 5
ProtocolType_1 = 5
ProtocolType_2 = 5
ProtocolType_3 = 5

; Channel selection by Phone number
ChannelSelectMode = 0

TrunkGroup = 0/1-31,1000,1
TrunkGroup = 1/1-31,2000,1
TrunkGroup = 2/1-31,3000,1
TrunkGroup = 3/1-31,4000,1
TrunkGroup = 0/16-16,7000,2
TrunkGroup = 1/16-16,7001,2
TrunkGroup = 2/16-16,7002,2
TrunkGroup = 3/16-16,7003,2

CoderName = 'g7231'
CoderName = 'Transparent'
CoderName_1 = 'g7231'
CoderName_2 = 'Transparent'
TelProfile_1 = voice,$$,1,$$,$$,$$,$$,$$,$$,$$
TelProfile_2 = data,$$,2,$$,$$,$$,$$,$$,$$,$$
```

13.3 QSIG Tunneling

The Mediant 1000 supports QSIG tunneling over SIP according to <draft-elwell-sipping-qsig-tunnel-03>. This method enables all QSIG messages to be sent as raw data in corresponding SIP messages using a dedicated message body. This mechanism is useful for two QSIG subscribers (connected to the same / different QSIG PBX) to communicate with each other over an IP network. Tunneling is supported for both directions (Tel to IP and IP to Tel).

The term tunneling means that messages are transferred 'as is' to the remote side, without being converted (QSIG→SIP→QSIG). The advantage of tunneling over QSIG→SIP interworking is that by using interworking, QSIG functionality can only be partially achieved. When tunneling is used, all QSIG capabilities are supported, whereas the tunneling medium (the SIP network) does not need to process these messages.

13.3.1 Implementation

QSIG messages are transferred in SIP messages in a separate Multipurpose Internet Mail Extensions (MIME) body. Therefore, if a message contains more than one body (e.g., SDP and QSIG), multipart MIME must be used. The Content-Type of the QSIG tunneled message is 'application/QSIG'. In addition, the Mediant 1000 adds a Content-Disposition header in the following format:

```
Content-Disposition: signal; handling=required.
```

- Call setup (originating gateway):
  The QSIG SETUP request is encapsulated in a SIP INVITE message without being altered. After the SIP INVITE request is sent, the gateway doesn't encapsulate the following QSIG message until a SIP 200 OK response is received. If the originating gateway receives a 4xx, 5xx or 6xx response, it disconnects the QSIG call with a 'no route to destination' cause.
- Call setup (terminating gateway):
  After the terminating gateway receives a SIP INVITE request with a Content-Type: application/QSIG, it sends the encapsulated QSIG SETUP message to the Tel side and sends a 200 OK response (no 1xx response is sent) to IP. The 200 OK response includes an encapsulated QSIG CALL PROCEEDING message (without waiting for a CALL PROCEEDING message from the Tel side). If tunneling is disabled and the incoming INVITE includes a QSIG body, a 415 response is sent.
- Mid-call communication:
  After the SIP connection is established, all QSIG messages are encapsulated in SIP INFO messages.
- Call tear-down:
  The SIP connection is terminated once the QSIG call is complete. The RELEASE COMPLETE message is encapsulated in the SIP BYE message that terminates the session.

To enable QSIG tunneling set the parameter EnableQSIGTunneling to 1 on both the originating and terminating gateways, and the parameter 'ISDNDuplicateQ931BuffMode' to 128 (duplicate all messages) (both parameters are described in Section 6.5.9 on page 237).

14 Security

This section describes the security mechanisms and protocols implemented on the Mediant 1000. The following list specifies the available security protocols and their objectives:

- IPSec and IKE protocols are part of the IETF standards for establishing a secured IP connection between two applications. IPSec and IKE are used in conjunction to provide security for control and management protocols but not for media (refer to Section 14.1 below).
- SSL (Secure Socket Layer) / TLS (Transport Layer Security) – The SSL / TLS protocols are used to provide privacy and data integrity between two communicating applications over TCP/IP. They are used to secure the following applications: SIP Signaling (SIPS), Web access (HTTPS) and Telnet access (refer to Section 14.2 on page 351).
- Secured RTP (SRTP) according to RFC 3711, used to encrypt RTP and RTCP transport (refer to Section 14.3 on page 356).
- RADIUS (Remote Authentication Dial-In User Service) - RADIUS server is used to enable multiple-user management on a centralized platform (refer to Section 14.4 on page 357).
- Internal Firewall allows filtering unwanted inbound traffic (refer to Section 14.5 on page 360).

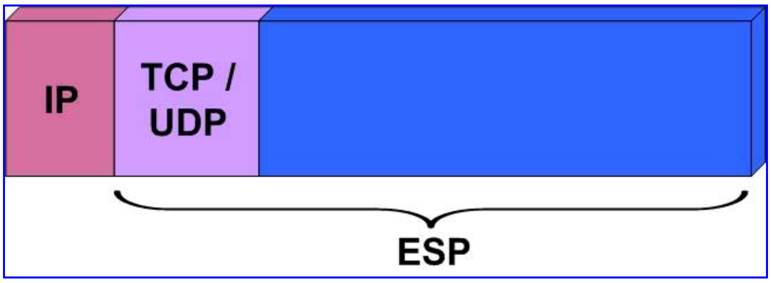
14.1 IPSec and IKE

IPSec and IKE protocols are part of the IETF standards for establishing a secured IP connection between two applications (also referred to as peers). Providing security services at the IP layer, IPSec and IKE are transparent to IP applications.

IPSec and IKE are used in conjunction to provide security for control and management (e.g., SNMP and Web) protocols but not for media (i.e., RTP, RTCP and T.38).

IPSec is responsible for securing the IP traffic. This is accomplished by using the Encapsulation Security Payload (ESP) protocol to encrypt the IP payload (illustrated in Figure 14-1 below). The IKE protocol is responsible for obtaining the IPSec encryption keys and encryption profile (known as IPSec Security Association (SA)).

Figure 14-1: IPSec Encryption



Note: IPSec doesn't function properly if the gateway's IP address is changed on-the-fly due to the fact that the crypto hardware can only be configured on reset. Therefore, reset the gateway after you change its IP address.

14.1.1 IKE

IKE is used to obtain the Security Associations (SA) between peers (the gateway and the application it's trying to contact). The SA contains the encryption keys and profile used by the IPSec to encrypt the IP stream. The IKE table lists the IKE peers with which the gateway performs the IKE negotiation (up to 20 peers are available).

The IKE negotiation is separated into two phases: main mode and quick mode. The main mode employs the Diffie-Hellman (DH) protocol to obtain an encryption key (without any prior keys), and uses a pre-shared key to authenticate the peers. The created channel secures the messages of the following phase (quick mode) in which the IPSec SA properties are negotiated.

The IKE negotiation is as follows:

- Main mode (the main mode creates a secured channel for the quick mode)
  - **SA negotiation:** The peers negotiate their capabilities using two proposals. Each proposal includes three parameters: Encryption method, Authentication protocol and the length of the key created by the DH protocol. The key's lifetime is also negotiated in this stage. For detailed information on configuring the main mode proposals, refer to Section 14.1.3.1 on page 345.
  - **Key exchange (DH):** The DH protocol is used to create a phase-1 key.
  - **Authentication:** The two peers authenticate one another using the pre-shared key (configured by the parameter 'IKEPolicySharedKey').
- Quick mode (quick mode negotiation is secured by the phase-1 SA)
  - **SA negotiation:** The peers negotiate their capabilities using a single proposal. The proposal includes two parameters: Encryption method and Authentication protocol. The lifetime is also negotiated in this stage. For detailed information on configuring the quick mode proposal, refer to the SPD table under Section 14.1.3.2 on page 348.
  - **Key exchange:** a symmetrical key is created using the negotiated SA.

IKE Specifications:

- Authentication mode: pre-shared key only
- Main mode is supported for IKE Phase 1
- Supported IKE SA encryption algorithms: Data Encryption Standard (DES), 3DES, and Advanced Encryption Standard (AES)
- Hash types for IKE SA: SHA1 and MD5

14.1.2 IPSec

IPSec is responsible for encrypting and decrypting the IP streams.

The IPSec Security Policy Database (SPD) table defines up to 20 IP peers to which the IPSec security is applied. IPSec can be applied to all packets designated to a specific IP address or to a specific IP address, port (source or destination) and protocol type.

Each outgoing packet is analyzed and compared to the SPD table. The packet's destination IP address (and optionally, destination port, source port and protocol type) are compared to each entry in the table. If a match is found, the gateway checks if an SA already exists for this entry. If it doesn't, the IKE protocol is invoked (refer to Section 14.1.1 above) and an IPSec SA is established. The packet is encrypted and transmitted. If a match isn't found, the packet is transmitted un-encrypted.

Note: An incoming packet whose parameters match one of the entries of the SPD table, but is received un-encrypted is dropped.

IPSec Specifications:

- Transport mode only
- Encapsulation Security Payload (ESP) only
- Support for Cipher Block Chaining (CBC)
- Supported IPSec SA encryption algorithms: DES, 3DES, and AES
- Hash types for IPSec SA include SHA1 and MD5

14.1.3 Configuring IPSec and IKE

To enable IPSec and IKE on the gateway set the ini file parameter 'EnableIPSec' to 1.

14.1.3.1 IKE Configuration

The parameters described in Table 14-1 below are used to configure the first phase (main mode) of the IKE negotiation for a specific peer. A different set of parameters can be configured for each of the 20 available peers.

Up to two IKE main mode proposals (Encryption / Authentication / DH group combinations) can be defined. The same proposals must be configured for all peers.

Table 14-1: IKE Table Configuration Parameters (continues on pages 345 to 346)

Parameter Name	Description
Shared Key **[IKEPolicySharedKey]**	Determines the pre-shared key (in textual format). Both peers must register the same pre-shared key for the authentication process to succeed. **Note 1:** The pre-shared key forms the basis of IPSec security and should therefore be handled cautiously (in the same way as sensitive passwords). It is not recommended to use the same pre-shared key for several connections. **Note 2:** Since the *ini* file is in plain text format, loading it to the gateway over a secure network connection is recommended, preferably over a direct crossed-cable connection from a management PC. For added confidentiality, use the encoded *ini* file option (described in Section 6.1 on page 213). **Note 3:** After it is configured, the value of the pre-shared key cannot be obtained via Web, *ini* file or SNMP (refer to Section 14.1.3.3 on page 351).
First to Fourth Proposal Encryption Type **[IKEPolicyProposalEncryption_X]**	Determines the encryption type used in the main mode negotiation for up to four proposals. X stands for the proposal number (0 to 3). The valid encryption values are: Not Defined (default) DES-CBC **[1]** Triple DES-CBC **[2]** AES **[3]**
First to Fourth Proposal Authentication Type **[IKEPolicyProposalAuthentication_X]**	Determines the authentication protocol used in the main mode negotiation for up to four proposals. X stands for the proposal number (0 to 3). The valid authentication values are: Not Defined (default) HMAC-SHA1-96) **[2]** HMAC-MD5-96 **[4]**
First to Fourth Proposal DH Group **[IKEPolicyProposalDHGroup_X]**	Determines the length of the key created by the DH protocol for up to four proposals. X stands for the proposal number (0 to 3). The valid DH Group values are: Not Defined (default) DH-786-Bit **[0]** DH-1024-Bit **[1]**
Authentication Method **[IkePolicyAuthenticationMethod]**	Determines the authentication method for IKE. The valid authentication method values include: 0 = Pre-shared Key (default) 1 = RSA Signiture **Note 1:** For pre-shared key based authentication, peers participating in an IKE exchange must have a prior (out-of-band) knowledge of the common key (see IKEPolicySharedKey parameter). **Note 2:** For RSA signature based authentication, peers must be loaded with a certificate signed by a common CA. For additional information on certificates, refer to Section 14.2.4 on page 353.

Table 14-1: IKE Table Configuration Parameters (continues on pages 345 to 346)

Parameter Name	Description
IKE SA LifeTime (sec) **[IKEPolicyLifeInSec]**	Determines the time (in seconds) the SA negotiated in the first IKE session (main mode) is valid. After the time expires, the SA is re-negotiated. The default value is 28800 (8 hours).
IKE SA LifeTime (KB) **[IKEPolicyLifeInKB]**	Determines the lifetime (in kilobytes) the SA negotiated in the first IKE session (main mode) is valid. After this size is reached, the SA is re-negotiated. The default value is 0 (this parameter is ignored).
The lifetime parameters (IKEPolicyLifeInSec and IKEPolicyLifeInKB) determine the duration the SA created in the main mode phase is valid. When the lifetime of the SA expires, it is automatically renewed by performing the IKE first phase negotiations. To refrain from a situation where the SA expires, a new SA is being negotiated while the old one is still valid. As soon as the new SA is created, it replaces the old one. This procedure occurs whenever an SA is about to expire.	

If no IKE methods are defined (Encryption / Authentication / DH Group), the default settings (shown in Table 14-2 below) are applied.

Table 14-2: Default IKE First Phase Proposals

	Encryption	Authentication	DH Group
Proposal 0	3DES	SHA1	1024
Proposal 1	3DES	MD5	1024
Proposal 2	3DES	SHA1	786
Proposal 3	3DES	MD5	786

➢ To configure the IKE table using the *ini* file:

The IKE parameters are configured using *ini* file tables (described in Section 12.5 on page 325). Each line in the table refers to a different IKE peer.

The Format line (IKE_DB_INDEX in the example below) specifies the order in which the actual data lines are written. The order of the parameters is irrelevant. Parameters are not mandatory unless stated otherwise. To support more than one Encryption / Authentication / DH Group proposals, for *each* proposal specify the relevant parameters in the Format line. Note that the proposal list must be contiguous.

Figure 14-2: Example of an IKE Table

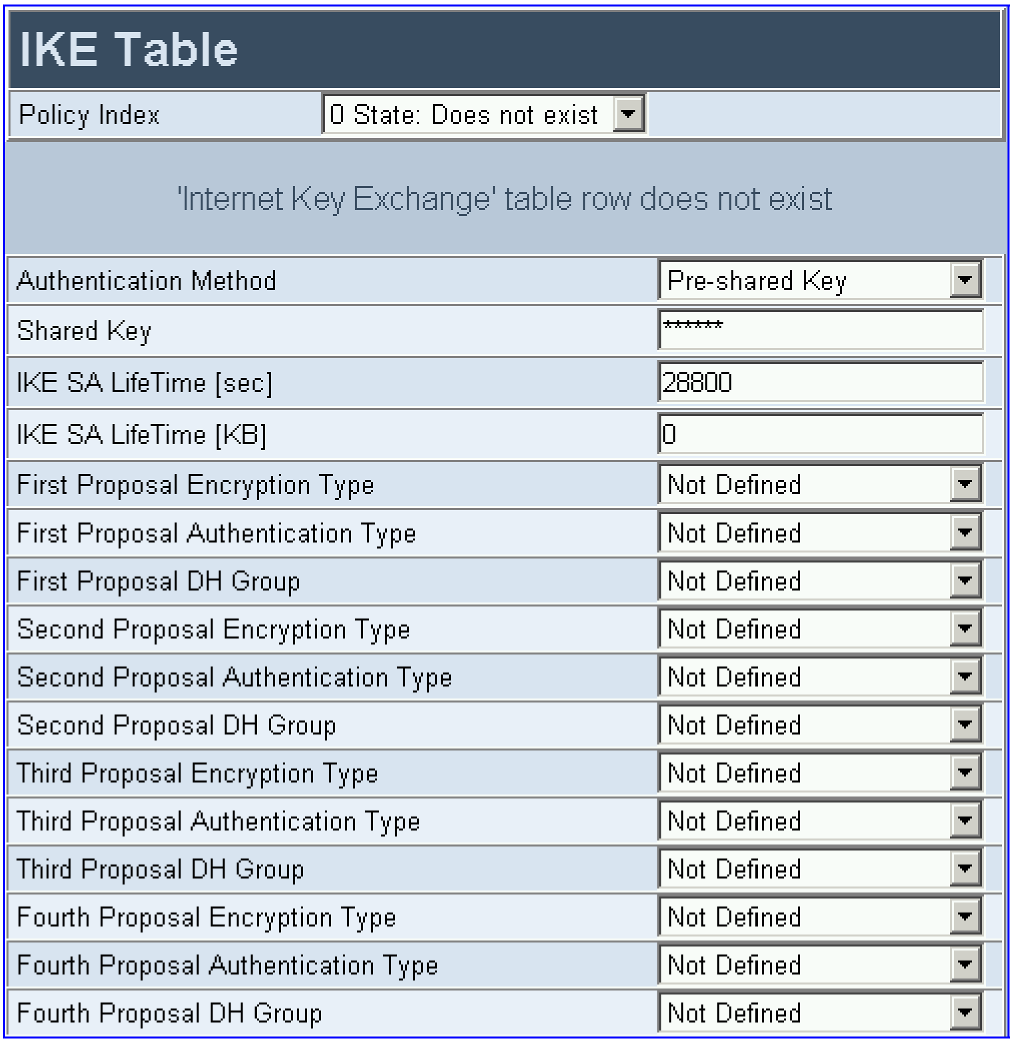
```
[IPSec_IKEDB_Table]

Format IKE_DB_INDEX = IKEPolicySharedKey, IKEPolicyProposalEncryption_0,
IKEPolicypRoposalAuthentication_0, IKEPolicyProposalDHGroup_0,
IKEPolicyProposalEncryption_1, IKEPolicypRoposalAuthentication_1,
IKEPolicyProposalDHGroup_1, IKEPolicyLifeInSec, IkePolicyAuthenticationMethod;

IPSEC_IKEDB_TABLE 0 = 123456789, 1, 2, 0, 2, 2, 1, 28800, 0;

[\IPSEC_IKEDB_TABLE]
```

In the example above, a single IKE peer is configured and a Pre-shared key authentication is selected. Its pre-shared key is 123456789. Two security proposals are configured: DES/SHA1/786DH and 3DES/SHA1/1024DH.

➢ To configure the IKE table using the Embedded Web Server, take these 6 steps:

1. Access the Embedded Web Server (refer to Section 5.3 on page 57).
2. Open the 'IKE Table' screen (**Advanced Configuration** menu > **Security Settings** > **IKE Table** option); the 'IKE Table' screen is displayed.

Figure 14-3: IKE Table Screen



IKE Table

Policy Index	0 State: Does not exist

'Internet Key Exchange' table row does not exist

Authentication Method	Pre-shared Key
Shared Key	******
IKE SA LifeTime [sec]	28800
IKE SA LifeTime [KB]	0
First Proposal Encryption Type	Not Defined
First Proposal Authentication Type	Not Defined
First Proposal DH Group	Not Defined
Second Proposal Encryption Type	Not Defined
Second Proposal Authentication Type	Not Defined
Second Proposal DH Group	Not Defined
Third Proposal Encryption Type	Not Defined
Third Proposal Authentication Type	Not Defined
Third Proposal DH Group	Not Defined
Fourth Proposal Encryption Type	Not Defined
Fourth Proposal Authentication Type	Not Defined
Fourth Proposal DH Group	Not Defined

3. In the 'Policy Index' drop-down list, select the peer you want to edit (up to 20 peers can be configured).
4. Configure the IKE parameters according to Table 14-1 on page 345.
5. Click the button **Create**; a row is create in the IKE table
6. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

To delete a peer from the IKE table, select it in the 'Policy Index' drop-down list, click the button **Delete** and click **OK** in the prompt.

14.1.3.2 IPSec Configuration

The parameters described in Table 14-3 below are used to configure the SPD table. A different set of parameters can be configured for each of the 20 available IP destinations.

Table 14-3: SPD Table Configuration Parameters (continues on pages 348 to 349)

<table>
<tr><th>Parameter Name</th><th colspan="2">Description</th></tr>
<tr><td>Remote IP Address
<b>[IPSecPolicyRemoteIPAddress]</b></td><td>Defines the destination IP address (or a FQDN) the IPSec mechanism is applied to.
This parameter is mandatory.
<b>Note:</b> When a FQDN is used, a DNS server must be configured (DNSPriServerIP).</td><td rowspan="5">IPSec is applied to outgoing packets whose IP address, destination port, source port and protocol type match the values defined for these four parameters.</td></tr>
<tr><td>Local IP Address Type
<b>[IPSecPolicyLocalIPAddressType]</b></td><td>Determines the local interface to which the encryption is applied (applicable to multiple IPs and VLANs).
0 = OAM interface (default).
1 = Control interface.</td></tr>
<tr><td>Source Port
<b>[IPSecPolicySrcPort]</b></td><td>Defines the source port the IPSec mechanism is applied to.
The default value is 0 (any port).</td></tr>
<tr><td>Destination Port
<b>[IPSecPolicyDstPort]</b></td><td>Defines the destination port the IPSec mechanism is applied to.
The default value is 0 (any port).</td></tr>
<tr><td>Protocol
<b>[IPSecPolicyProtocol]</b></td><td>Defines the protocol type the IPSec mechanism is applied to.
0 = Any protocol (default).
17 = UDP.
6 = TCP.
Or any other protocol type defined by IANA (Internet Assigned Numbers Authority).</td></tr>
<tr><td>Related Key Exchange Method Index
<b>[IPsecPolicyKeyExchangeMethodIndex]</b></td><td colspan="2">Determines the index for the corresponding IKE entry. Note that several policies can be associated with a single IKE entry.
The valid range is 0 to 19. The default value is 0.</td></tr>
<tr><td colspan="3"><b>IKE Second Phase Parameters (Quick Mode)</b></td></tr>
<tr><td>SA Lifetime (sec)
<b>I[PsecPolicyLifeInSec]</b></td><td colspan="2">Determines the time (in seconds) the SA negotiated in the second IKE session (quick mode) is valid. After the time expires, the SA is re-negotiated.
The default value is 28800 (8 hours).</td></tr>
<tr><td>SA Lifetime (KB)
<b>[IPSecPolicyLifeInKB]</b></td><td colspan="2">Determines the lifetime (in kilobytes) the SA negotiated in the second IKE session (quick mode) is valid. After this size is reached, the SA is re-negotiated.
The default value is 0 (this parameter is ignored).</td></tr>
<tr><td colspan="3">The lifetime parameters (IPsecPolicyLifeInSec and IPSecPolicyLifeInKB) determine the duration of which an SA is valid. When the lifetime of the SA expires, it is automatically renewed by performing the IKE second phase negotiations. To refrain from a situation where the SA expires, a new SA is being negotiated while the old one is still valid. As soon as the new SA is created, it replaces the old one. This procedure occurs whenever an SA is about to expire.</td></tr>
<tr><td>First to Fourth Proposal Encryption Type
<b>[IPSecPolicyProposalEncryption_X]</b></td><td colspan="2">Determines the encryption type used in the quick mode negotiation for up to four proposals.
X stands for the proposal number (0 to 3).
The valid encryption values are:
Not Defined (default)
None <b>[0]</b> = No encryption
DES-CBC <b>[1]</b>
Triple DES-CBC <b>[2]</b>
AES <b>[3]</b></td></tr>
</table>

Table 14-3: SPD Table Configuration Parameters (continues on pages 348 to 349)

Parameter Name	Description
First to Fourth Proposal Authentication Type **[IPSecPolicyProposalAuthentication_X]**	Determines the authentication protocol used in the quick mode negotiation for up to four proposals. X stands for the proposal number (0 to 3). The valid authentication values are: Not Defined (default) HMAC-SHA-1-96 **[2]** HMAC-MD5-96 **[4]**

If no IPSec methods are defined (Encryption / Authentication), the default settings (shown in Table 14-4 below) are applied.

Table 14-4: Default IKE Second Phase Proposals

	Encryption	Authentication
Proposal 0	3DES	SHA1
Proposal 1	3DES	MD5
Proposal 2	DES	SHA1
Proposal 3	DES	MD5

➢ To configure the SPD table using the *ini* file:

SPD table is configured using *ini* file tables (described in Section 12.5 on page 325). Each line in the table refers to a different IP destination.

The Format line (SPD_INDEX in the example below) specifies the order in which the actual data lines are written. The order of the parameters is irrelevant. Parameters are not mandatory unless stated otherwise. To support more than one Encryption / Authentication proposals, for *each* proposal specify the relevant parameters in the Format line. Note that the proposal list must be contiguous.

Figure 14-4: Example of an SPD Table

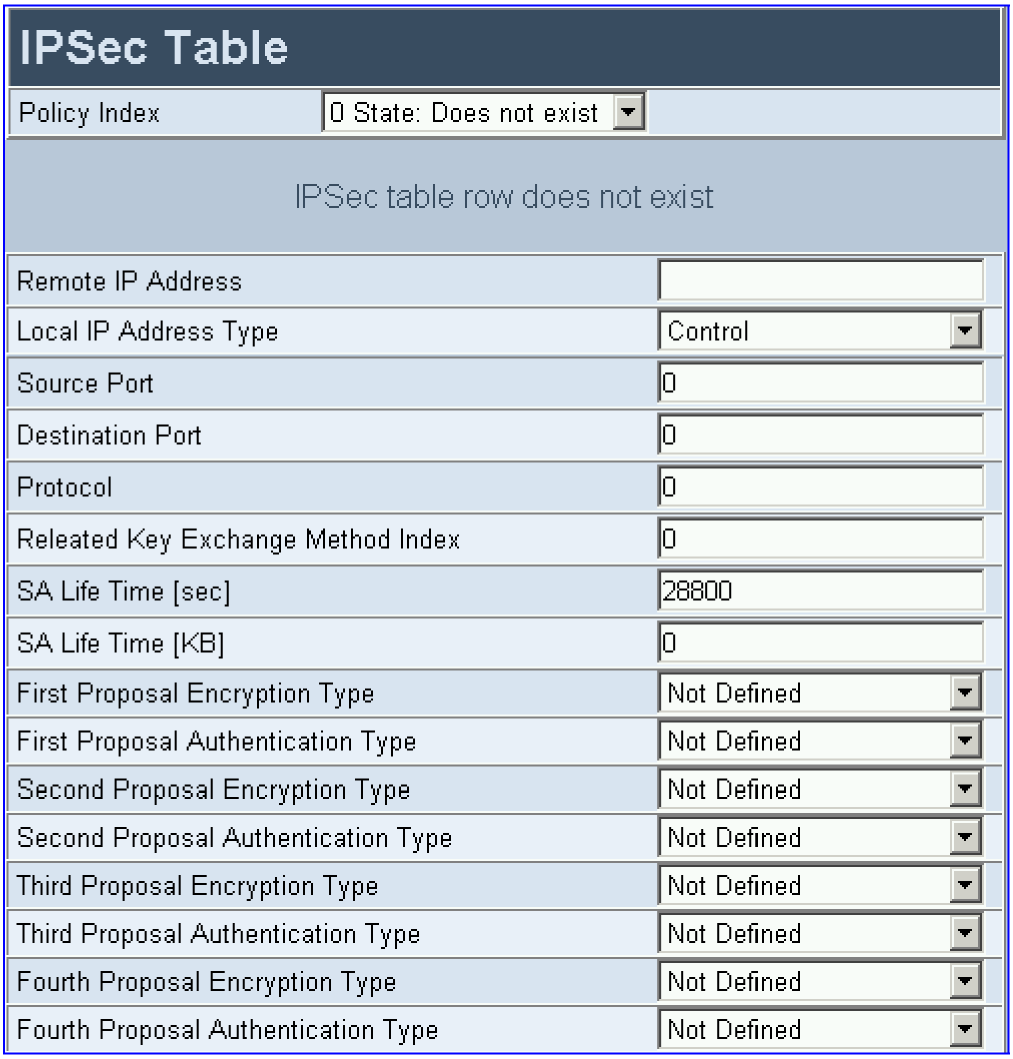
```
[ IPSEC_SPD_TABLE ]

Format SPD_INDEX = IPSecPolicyRemoteIPAddress, IpsecPolicySrcPort,
IPSecPolicyDStPort,IPSecPolicyProtocol, IPSecPolicyLifeInSec,
IPSecPolicyProposalEncryption_0, IPSecPolicyProposalAuthentication_0,
IPSecPolicyProposalEncryption_1, IPSecPolicyProposalAuthentication_1,
IPSecPolicyKeyExchangeMethodIndex, IPSecPolicyLocalIPAddressType;

IPSEC_SPD_TABLE 0 = 10.11.2.21, 0, 0, 17, 900, 1,2, 2,2 ,1, 0;

[ \IPSEC_SPD_TABLE ]
```

In the SPD example, all packets designated to IP address 10.11.2.21 that originates from the OAM interface (regardless to their destination and source ports) and whose protocol is UDP are encrypted, the SPD also defines an SA lifetime of 900 seconds and two security proposals: DES/SHA1 and 3DES/SHA1.

➢ To configure the SPD table using the Embedded Web Server, take these 6 steps:

1. Access the Embedded Web Server (refer to Section 5.3 on page 57).
2. Open the 'IPSec Table' screen (**Advanced Configuration** menu > **Security Settings** > **IPSec Table** option); the 'IPSec Table' screen is displayed.

Figure 14-5: IPSec Table Screen



IPSec Table

Policy Index: 0 State: Does not exist

IPSec table row does not exist

Remote IP Address	
Local IP Address Type	Control
Source Port	0
Destination Port	0
Protocol	0
Releated Key Exchange Method Index	0
SA Life Time [sec]	28800
SA Life Time [KB]	0
First Proposal Encryption Type	Not Defined
First Proposal Authentication Type	Not Defined
Second Proposal Encryption Type	Not Defined
Second Proposal Authentication Type	Not Defined
Third Proposal Encryption Type	Not Defined
Third Proposal Authentication Type	Not Defined
Fourth Proposal Encryption Type	Not Defined
Fourth Proposal Authentication Type	Not Defined

3. In the 'Policy Index' drop-down list, select the rule you want to edit (up to 20 rules can be configured).
4. Configure the SPD parameters according to Table 14-3 on page 348.
5. Click the button **Create**; a row is create in the SPD table
6. To save the changes so they are available after a power fail, refer to Section 5.9.2 on page 200.

To delete a peer from the SPD table, select it in the 'Policy Index' drop-down list, click the button **Delete** and click **OK** in the prompt.

14.1.3.3 IPSec and IKE Configuration Table’s Confidentiality

Since the pre-shared key parameter of the IKE table must remain undisclosed, measures are taken by the *ini* file, Embedded Web Server and SNMP agent to maintain this parameter’s confidentiality. On the Embedded Web Server a list of asterisks is displayed instead of the pre-shared key. On SNMP, the pre-shared key parameter is a write-only parameter and cannot be read. In the *ini* file, the following measures to assure the secrecy of the IPSec and IKE tables are taken:

- **Hidden IPSec and IKE tables:** When uploading the *ini* file from the gateway, the IPSec and IKE tables are not available. Instead, the notifications (shown in Figure 14-6) are displayed.

Figure 14-6: Example of an *ini* File Notification of Missing Tables

```
;
;  *** TABLE IPSEC_IKEDB_TABLE ***
; This table contains hidden elements and will not be exposed.
; This table exists on board and will be saved during restarts
;


;
;  *** TABLE IPSEC_SPD_TABLE ***
; This table contains hidden elements and will not be exposed.
; This table exists on board and will be saved during restarts
;
```

- **Preserving the values of the parameters in the IPSec and IKE tables from one *ini* file loading to the next:** The values configured for the parameters in the IPSec tables in the *ini* file are preserved from one loading to another. If a newly loaded *ini* file doesn’t define IPSec tables, the previously loaded tables remain valid. To invalidate a previously loaded *ini* file's IPSec tables, load a new *ini* file with an empty IPSec table (shown below).

Figure 14-7: Empty IPSec / IKE Tables

```
[IPSec_IKEDB_Table]
[\IPSec_IKEDB_Table]

[IPSEC_SPD_TABLE]
[\IPSEC_SPD_TABLE]
```

14.2 SSL/TLS

SSL, also known as TLS, is the method used to secure the Mediant 1000 SIP Signaling connections, Embedded Web Server and Telnet server. The SSL protocol provides confidentiality, integrity and authenticity between two communicating applications over TCP/IP.

Specifications for the SSL/TLS implementation include the following:

- Supports transports: SSL 2.0, SSL 3.0, TLS 1.0
- Supports ciphers: DES, RC4 compatible
- Authentication: X.509 certificates; CRLs are not supported

14.2.1 SIP Over TLS (SIPS)

The Mediant 1000 uses TLS over TCP to encrypt SIP transport and (optionally) to authenticate it. To enable TLS on the Mediant 1000, set the selected transport type to TLS (SIPTransportType = 2). In this mode the gateway initiates a TLS connection only for the next network hop. To enable TLS all the way to the destination (over multiple hops) set EnableSIPS to 1. When a TLS connection with the gateway is initiated, the gateway also responds using TLS regardless of the configured SIP transport type (in this case, the parameter EnableSIPS is also ignored).

TLS and SIPS use the Certificate Exchange process described in Sections 14.2.4 and 14.2.5. To change the port number used for SIPS transport (by default 5061), use the parameter TLSLocalSIPPort.

When SIPS is used, it is sometimes required to use two-way authentication. When acting as the TLS server (in a specific connection) it is possible to demand the authentication of the client's certificate. To enable two-way authentication on the Mediant 1000, set the ini file parameter, SIPSRequireClientCertificate = 1. For information on installing a client certificate, refer to Section 14.2.5 on page 355.

14.2.2 Embedded Web Server Configuration

For additional security, you can configure the Embedded Web Server to accept only secured (HTTPS) connections by changing the parameter HTTPSOnly to 1 (described in Table 5-36 on page 176).

You can also change the port number used for the secured Web server (by default 443), by changing the *ini* file parameter, HTTPSPort (described in Section 6.5.4 on page 227).

14.2.2.1 Using the Secured Embedded Web Server

➢ To use the secured Embedded Web Server, take these 3 steps:

1. Access the Mediant 1000 using the following URL: https://[host name] or [IP address]

   Depending on the browser's configuration, a security warning dialog may be displayed. The reason for the warning is that the Mediant 1000 initial certificate is not trusted by your PC. The browser may allow you to install the certificate, thus skipping the warning dialog the next time you connect to the Mediant 1000.

2. If you are using Internet Explorer, click **View Certificate** and then **Install Certificate**.

3. The browser also warns you if the host name used in the URL is not identical to the one listed in the certificate. To solve this, add the IP address and host name (ACL_nnnnnn where nnnnnn is the serial number of the Mediant 1000) to your hosts file, located at /etc/hosts on UNIX or C:\Windows\System32\Drivers\ETC\hosts on Windows; then use the host name in the URL (e.g., https://ACL_280152).The figure below is an example of a host file:

Figure 14-8: Example of a Host File

```
# This is a sample HOSTS file used by Microsoft TCP/IP for Windows.
# Location: C:\WINDOWS\SYSTEM32\DRIVERS\ETC\hosts
#
127.0.0.1     localhost
10.31.4.47    ACL_280152
```

14.2.3 Secured Telnet

To enable the embedded Telnet server on the Mediant 1000, set the parameter TelnetServerEnable (described in Table 5-24 on page 142) to 1 (standard mode) or 2 (SSL mode); no information is transmitted in the clear when SSL mode is used.

If the Telnet server is set to SSL mode, a special Telnet client is required on your PC to connect to the Telnet interface over a secured connection; examples include C-Kermit for UNIX, Kermit-95 for Windows, and AudioCodes' acSSLTelnet utility for Windows (that requires prior installation of the free OpenSSL toolkit). Contact AudioCodes to obtain the acSSLTelnet utility.

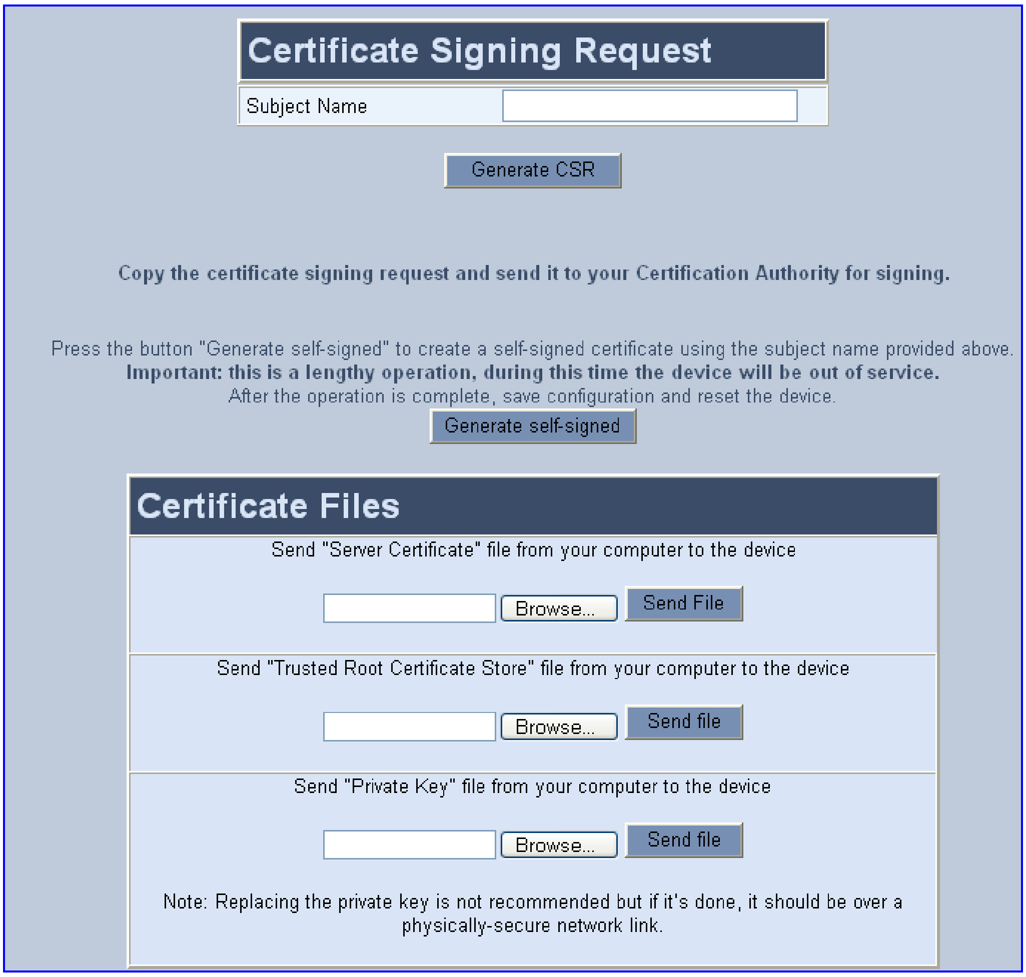
14.2.4 Server Certificate Replacement

The Mediant 1000 is supplied with a working SSL configuration consisting of a unique self-signed server certificate. If an organizational Public Key Infrastructure (PKI) is used, you may wish to replace this certificate with one provided by your security administrator.

➢ **To replace the Mediant 1000 self-signed certificate, take these 9 steps:**

1. Your network administrator should allocate a unique DNS name for the Mediant 1000 (e.g., dns_name.corp.customer.com). This name is used to access the device, and should therefore be listed in the server certificate.
2. Open the ‘Certificates’ screen (**Advanced Configuration** menu > **Security Settings** submenu > **Certificates** option); the ‘Certificates’ screen is displayed (Figure 14-9).

Figure 14-9: Certificate Signing Request Screen



Copy the certificate signing request and send it to your Certification Authority for signing.

Press the button "Generate self-signed" to create a self-signed certificate using the subject name provided above.
Important: this is a lengthy operation, during this time the device will be out of service.
After the operation is complete, save configuration and reset the device.

Send "Server Certificate" file from your computer to the device

Send "Trusted Root Certificate Store" file from your computer to the device

Send "Private Key" file from your computer to the device

Note: Replacing the private key is not recommended but if it's done, it should be over a physically-secure network link.

3. In the Subject Name field, enter the DNS name and click **Generate CSR**. A textual certificate signing request, that contains the SSL device identifier, is displayed.
4. Copy this text and send it to your security provider; the security provider (also known as Certification Authority or CA) signs this request and send you a server certificate for the device.
5. Save the certificate in a file (e.g., cert.txt). Ensure the file is a plain-text file with the 'BEGIN CERTIFICATE' header. The figure below is an example of a Base64-Encoded X.509 Certificate.

Figure 14-10: Example of a Base64-Encoded X.509 Certificate

```
-----BEGIN CERTIFICATE-----
MIIDkzCCAnugAwIBAgIEAgAAADANBgkqhkiG9w0BAQQFADA/MQswCQYDVQQGEwJG
UjETMBEGA1UEChMKQ2VydGlwb3N0ZTEbMBkGA1UEAxMSQ2VydGlwb3N0ZSBTZXJ2
ZXVyMB4XDTk4MDYyNDA4MDAwMFoXDTE4MDYyNDA4MDAwMFowPzELMAkGA1UEBhMC
RlIxEzARBgNVBAoTCkNlcnRpcG9zdGUxGzAZBgNVBAMTEkNlcnRpcG9zdGUgU2Vy
dmV1cjCCASEwDQYJKoZIhvcNAQEBBQADggEOADCCAQkCggEAPqd4MziR4spWldGR
x8bQrhZkonWnNm`+Yhb7+4Q67ecf1janH7GcN/SXsfx7jJpreWULf7v7Cvpr4R7qI
JcmdHIntmf7JPM5n6cDBv17uSW63er7NkVnMFHwK1QaGFLMybFkzaeGrvFm4k3lR
efiXDmuOe+FhJgHYezYHf44LvPRPwhSrzi9+Aq3o8pWDguJuZDIUP1F1jMa+LPwv
REXfFcUW+w==
-----END CERTIFICATE-----
```

6. Before continuing, set the parameter HTTPSOnly = 0 to ensure you have a method of accessing the device in case the new certificate doesn't work. Restore the previous setting after testing the configuration.
7. In the Certificates screen (Figure 14-9) locate the server certificate loading section.
8. Click **Browse**, navigate to the cert.txt file, and then click **Send File**.
9. When the operation is completed, save the configuration (Section 5.9.2 on page 200) and restart the Mediant 1000; the Embedded Web Server uses the provided certificate.

Notes:

- The certificate replacement process can be repeated when necessary (e.g., the new certificate expires).
- It is possible to use the IP address of the Mediant 1000 (e.g., 10.3.3.1) instead of a qualified DNS name in the Subject Name. This practice is not recommended since the IP address is subject to changes and may not uniquely identify the device.
- The server certificate can also be loaded via *ini* file using the parameter 'HTTPSCertFileName'.

14.2.5 Client Certificates

By default, Web servers using SSL provide one-way authentication. The client is certain that the information provided by the Web server is authentic. When an organizational PKI is used, two-way authentication may be desired: both client and server should be authenticated using X.509 certificates. This is achieved by installing a client certificate on the managing PC, and loading the same certificate (in base64-encoded X.509 format) to the Mediant 1000 Trusted Root Certificate Store. The Trusted Root Certificate file should contain both the certificate of the authorized user and the certificate of the CA.

Since X.509 certificates have an expiration date and time, the Mediant 1000 must be configured to use NTP (Section 10.8 on page 308) to obtain the current date and time. Without a correct date and time, client certificates cannot work.

➢ To install a client certificate, take these 6 steps:

1. Before continuing, set HTTPSOnly = 0 to ensure you have a method of accessing the device in case the client certificate doesn't work. Restore the previous setting after testing the configuration.
2. Open the 'Certificates' screen (**Advanced Configuration** menu > **Security Settings** submenu > **Certificates** option); the 'Certificates' screen is displayed (Figure 14-9).
3. To load the Trusted Root Certificate file locate the trusted root certificate loading section.
4. Click **Browse** and navigate to the file, click **Send File**.
5. When the operation is completed, set the ini file parameter, HTTPSRequireClientCertificates = 1.
6. Save the configuration (Section 5.9.2 on page 200) and restart the Mediant 1000.

When a user connects to the secure Web server:

- If the user has a client certificate from a CA listed in the Trusted Root Certificate file, the connection is accepted and the user is prompted for the system password.
- If both the CA certificate and the client certificate appear in the Trusted Root Certificate file, the user is not prompted for a password (thus providing a single-sign-on experience - the authentication is performed using the X.509 digital signature).
- If the user doesn't have a client certificate from a listed CA, or doesn't have a client certificate at all, the connection is rejected.

Notes:

- The process of installing a client certificate on your PC is beyond the scope of this document. For more information, refer to your Web browser or operating system documentation, and/or consult your security administrator.
- The root certificate can also be loaded via *ini* file using the parameter 'HTTPSRootFileName'.

14.3 SRTP

The gateway supports Secured RTP (SRTP) according to RFC 3711. SRTP is used to encrypt RTP and RTCP transport since it is best-suited for protecting VoIP traffic.

SRTP requires a Key Exchange mechanism that is performed according to <draft-ietf-mmusic-sdescriptions-12>. The Key Exchange is executed by adding a 'Crypto' attribute to the SDP. This attribute is used (by both sides) to declare the various supported cipher suites and to attach the encryption key to use. If negotiation of the encryption data is successful, the call is established.

Use the parameter MediaSecurityBehaviour (described in Section 5.6.7.5 on page 175) to select the gateway's mode of operation that determine the behavior of the gateway if negotiation of the cipher suite fails:

- Mandatory: the call is terminated. Incoming calls that don't include encryption information are rejected.
- Preferable: an unencrypted call is established. Incoming calls that don't include encryption information are accepted.

To enable SRTP, set the parameter EnableMediaSecurity to 1 (described in Section 5.6.7.5 on page 175).

Notes:

- When SRTP is used, the channel capacity is reduced (refer to the parameter EnableMediaSecurity).
- The gateway only supports the AES 128 in CM mode cipher suite.

Figure 14-11: Example of crypto Attributes Usage

```
a=crypto:1 AES_CM_128_HMAC_SHA1_80 inline:PsKoMpHlCg+b5X0YLuSvNrImEh/dAe
a=crypto:2 AES_CM_128_HMAC_SHA1_32 inline:IsPtLoGkBf9a+c6XVzRuMqHlDnEiAd
```

14.4 RADIUS Login Authentication

Users can enhance the security and capabilities of logging to the gateway's Web and Telnet embedded servers by using a Remote Authentication Dial-In User Service (RADIUS) to store numerous usernames, passwords and access level attributes (Web only), allowing multiple user management on a centralized platform. RADIUS (RFC 2865) is a standard authentication protocol that defines a method for contacting a predefined server and verifying a given name and password pair against a remote database, in a secure manner.

When accessing the Web and Telnet servers, users must provide a valid username and password. When RADIUS authentication isn't used, the username and password are authenticated with the Embedded Web Server's usernames and passwords of the primary or secondary accounts (refer to Section 5.2.1 on page 55) or with the Telnet server's username and password stored internally in the gateway's memory. When RADIUS authentication is used, the gateway doesn't store the username and password but simply forwards them to the pre-configured RADIUS server for authentication (acceptance or rejection). The internal Web / Telnet passwords can be used as a fallback mechanism in case the RADIUS server doesn't respond (configured by the parameter BehaviorUponRadiusTimeout). Note that when RADIUS authentication is performed, the Web / Telnet servers are blocked until a response is received (with a timeout of 5 seconds).

RADIUS authentication requires HTTP basic authentication, meaning the username and password are transmitted in clear text over the network. Therefore, users are recommended to set the parameter 'HttpsOnly = 1' to force the use of HTTPS, since the transport is encrypted.

14.4.1 Setting Up a RADIUS Server

The following examples refer to FreeRADIUS, a free RADIUS server that can be downloaded from www.freeradius.org. Follow the directions on that site for information on installing and configuring the server. If you use a RADIUS server from a different vendor, refer to its appropriate documentation.

➢ To set up a RADIUS server, take these 5 steps:

1. Define the gateway as an authorized client of the RADIUS server, with a predefined 'shared secret' (a password used to secure communication) and a vendor ID. The figure below displays an example of the file clients.conf (FreeRADIUS client configuration).

Figure 14-12: Example of the File clients.conf (FreeRADIUS Client Configuration)

```
#
# clients.conf - client configuration directives
#
client 10.31.4.47 {
        secret          = FutureRADIUS
        shortname       = tp1610_master_tpm
}
```

2. If access levels are required, set up a VSA dictionary for the RADIUS server and select an attribute ID that represents each user's access level. The following example shows a dictionary file for FreeRADIUS that defines the attribute ‘ACL-Auth-Level’ with ID=35.

Figure 14-13: Example of a Dictionary File for FreeRADIUS (FreeRADIUS Client Configuration)

```
#
# AudioCodes VSA dictionary
#
VENDOR AudioCodes 5003
ATTRIBUTE ACL-Auth-Level 35 integer AudioCodes
VALUE ACL-Auth-Level ACL-Auth-UserLevel 50
VALUE ACL-Auth-Level ACL-Auth-AdminLevel 100
VALUE ACL-Auth-Level ACL-Auth-SecurityAdminLevel 200
```

3. In the RADIUS server, define the list of users authorized to use the gateway, using one of the password authentication methods supported by the server implementation. The following example shows a user configuration file for FreeRADIUS using a plain-text password.

Figure 14-14: Example of a User Configuration File for FreeRADIUS Using a Plain-Text Password

```
# users - local user configuration database

john    Auth-Type := Local, User-Password == "qwerty"
        Service-Type = Login-User,
        ACL-Auth-Level = ACL-Auth-SecurityAdminLevel

larry   Auth-Type := Local, User-Password == "123456"
        Service-Type = Login-User,
        ACL-Auth-Level = ACL-Auth-UserLevel
```

4. Record and retain the IP address, port number, ’shared secret’, vendor ID and VSA access level identifier (if access levels are used) used by the RADIUS server.
5. Configure the gateway’s relevant parameters according to Section 14.4.2.

14.4.2 Configuring RADIUS Support

For information on the RADIUS parameters, refer to Table 5-36 on page 176.

➢ To configure RADIUS support on the gateway via the Embedded Web Server, take these 13 steps:

1. Access the Embedded Web Server (refer to Section 5.3 on page 57).
2. Open the ‘General Security Settings’ screen (**Advanced Configuration** menu > **Security Settings** > **General Security Settings** option); the ‘General Security Settings’ screen is displayed.
3. Under section ‘General RADIUS Settings’, in the field ‘Enable RADIUS Access Control’, select ‘Enable’; the RADIUS application is enabled.
4. In the field ‘Use RADIUS for Web / Telnet Login’, select ‘Enable’; RADIUS authentication is enabled for Web and Telnet login.
5. Enter the RADIUS server IP address, port number and shared secret in the relevant fields.

6. Under section ‘RADIUS Authentication Settings’, in the field ‘Device Behavior Upon RADIUS Timeout’, select the gateway’s operation if a response isn’t received from the RADIUS server after the 5 seconds timeout expires:

   - Deny Access – the gateway denies access to the Web and Telnet embedded servers.
   - Verify Access Locally – the gateway checks the local username and password.

7. In the field ‘Local RADIUS Password Cache Timeout’, enter a time (in seconds); when this time expires, the username and password verified by the RADIUS server becomes invalid and a username and password must be re-validated with the RADIUS server.

8. In the field ‘Local RADIUS Password Cache Mode’, select the gateway’s mode of operation regarding the above-mentioned ‘Local RADIUS Password Cache Timer’ option:

   - Reset Timer Upon Access – upon each access to a Web screen, the timer resets (reverts to the initial value configured in the previous step).
   - Absolute Expiry Timer - when you access a Web screen, the timer doesn’t reset but rather continues decreasing.

9. In the field ‘RADIUS VSA Vendor ID’, enter the vendor ID you configured in the RADIUS server:

10. When using the Web access-level mechanism, perform one of the following options:

    - When RADIUS responses include the access level attribute:
      In the field ‘RADIUS VSA Access Level Attribute’, enter the code that indicates the access level attribute in the Vendor Specific Attributes (VSA) section of the received RADIUS packet.
    - When RADIUS responses don’t include the access level attribute:
      In the field ‘Default Access Level’, enter the default access level that is applied to all users authenticated by the RADIUS server.

11. In the field ‘Require Secured Web Connection (HTTPS)’, select ‘HTTPS only’. It is important you use HTTPS (secure Web server) when connecting to the gateway over an open network, since the password is transmitted in clear text. Similarly, for Telnet, use SSL ‘TelnetServerEnable = 2 (refer to Section 14.2.3 on page 353).

12. Save the changes so they are available after a power fail.

13. Reset the gateway (refer to Section 5.9.3 on page 200).

After reset, when accessing the Web or Telnet servers, use the username and password you configured in the RADIUS database. The local system password is still active and can be used when the RADIUS server is down.

➢ To configure RADIUS support on the gateway using the *ini* file:

- Add the following parameters to the *ini* file. For information on modifying the *ini* file, refer to Section 6.2 on page 213.
  - EnableRADIUS = 1
  - WebRADIUSLogin = 1
  - RADIUSAuthServerIP = IP address of RADIUS server
  - RADIUSAuthPort = port number of RADIUS server, usually 1812
  - SharedSecret = your shared secret'
  - HTTPSOnly = 1
  - BehaviorUponRadiusTimeout = 1
  - RadiusLocalCacheMode = 1
  - RadiusLocalCacheTimeout = 300
  - RadiusVSAVendorID = *your vendor’s ID*
  - RadiusVSAAccessAttribute = *code that indicates the access level attribute*
  - DefaultAccessLevel = *default access level (0 to 200)*

14.5 Internal Firewall

The Mediant 1000 accommodates an internal access list facility, allowing the security administrator to define network traffic filtering rules. The access list provides the following features:

- Block traffic from known malicious sources
- Only allow traffic from known friendly sources, and block all others
- Mix allowed and blocked network sources
- Limit traffic to a predefined rate (blocking the excess)
- Limit traffic to specific protocols, and specific port ranges on the device

The access list consists of a table with up to 50 ordered lines. For each packet received on the network interface, the table is scanned from the top until a matching rule is found (or the table end is reached). This rule can either block the packet or allow it; however it is important to note that subsequent rules aren't scanned. If the table end is reached without a match, the packet is accepted.

Each rule is composed of the following fields (described in Table 5-35 on page 174):

- IP address (or DNS name) of source network
- IP network mask
- Destination UDP/TCP ports (on this device)
- Protocol type
- Maximum packet size, byte rate per second, and allowed data burst
- Action upon match (allow or block)

Figure 14-15 shows an example of an access list definition via *ini* file:

Figure 14-15: Example of an Access List Definition via *ini* File

```
[ ACCESSLIST ]
FORMAT AccessList_Index = AccessList_Source_IP, AccessList_Net_Mask,
AccessList_Start_Port, AccessList_End_Port, AccessList_Protocol,
AccessList_Packet_Size, AccessList_Byte_Rate, AccessList_Byte_Burst,
AccessList_Allow_Type;

AccessList 10 = mgmt.customer.com, 255.255.255.255, 0, 80, tcp, 0, 0, 0, allow ;
AccessList 15 = 192.0.0.0, 255.0.0.0, 0, 65535, any, 0, 40000, 50000, block ;
AccessList 20 = 10.31.4.0, 255.255.255.0, 4000, 9000, any, 0, 0, 0, block ;
AccessList 22 = 10.4.0.0, 255.255.0.0, 4000, 9000, any, 0, 0, 0, block ;
[ \ACCESSLIST ]
```

Explanation of the example access list:

- Rule #10: traffic from the host 'mgmt.customer.com' destined to TCP ports 0 to 80, is always allowed.
- Rule #15: traffic from the 192.xxx.yyy.zzz subnet, is limited to a rate of 40 Kbytes per second (with an allowed burst of 50 Kbytes). Note that the rate is specified in bytes, not bits, per second; a rate of 40000 bytes per second, nominally corresponds to 320 kbps.
- Rule #20: traffic from the subnet 10.31.4.xxx destined to ports 4000 to 9000 is always blocked, regardless of protocol.
- Rule #22: traffic from the subnet 10.4.xxx.yyy destined to ports 4000 to 9000 is always blocked, regardless of protocol.
- All other traffic is allowed.

More complex rules may be defined, relying on the 'single-match' process described above:

Figure 14-16 shows an advanced example of an access list definition via *ini* file:

Figure 14-16: Advanced Example of an Access List Definition via *ini* File

```
[ ACCESSLIST ]
FORMAT AccessList_Index = AccessList_Source_IP, AccessList_Net_Mask,
AccessList_Start_Port, AccessList_End_Port, AccessList_Protocol,
AccessList_Packet_Size, AccessList_Byte_Rate, AccessList_Byte_Burst,
AccessList_Allow_Type;

AccessList 10 = 10.0.0.0, 255.0.0.0, 0, 65535, any, 0, 40000, 50000, allow ;
AccessList 15 = 10.31.4.0, 255.255.255.0, 4000, 9000, any, 0, 0, 0, allow ;
AccessList 20 = 0.0.0.0, 0.0.0.0, 0, 65535, any, 0, 0, 0, block;
 [ \ACCESSLIST ]
```

Explanation of the example access list:

This access list consists of three rules:

- Rule #10: traffic from the subnet 10.xxx.yyy.zzz is allowed if the traffic rate does not exceed 40 KB/s.
- Rule #15: if a packet didn't match rule #10, that is, the excess traffic is over 40 KB/s, and coming from the subnet 10.31.4.xxx to ports 4000 to 9000, then it is allowed.
- Rule #20: all other traffic (which didn't match the previous rules), is blocked.

The internal firewall can also be configured via the Embedded Web Server (refer to Section 5.6.7.3 on page 172).

14.6 Network Port Usage

The following table lists the default TCP/UDP network port numbers used by the Mediant 1000. Where relevant, the table lists the ini file parameters that control the port usage and provide source IP address filtering capabilities.

Table 14-5: Default TCP/UDP Network Port Numbers

Port Number	Peer Port	Application	Notes
2	2	Debugging interface	Always ignored
23	-	Telnet	Disabled by default (TelnetServerEnable). Configurable (TelnetServerPort), access controlled by WebAccessList
68	67	DHCP	Active only if DHCPEnable = 1
80	-	Web server (HTTP)	Configurable (HTTPPort), can be disabled (DisableWebTask or HTTPSOnly). Access controlled by WebAccessList
161	-	SNMP GET/SET	Configurable (SNMPPort), can be disabled (DisableSNMP). Access controlled by SNMPTrustedMGR
443	-	Web server (HTTPS)	Configurable (HTTPSPort), can be disabled (DisableWebTask). Access controlled by WebAccessList
500	-	IPSec IKE	Can be disabled (EnableIPSec)
6000, 6010 and up	-	RTP traffic	Base port number configurable (BaseUDPPort), fixed increments of 10. The number of ports used depends on the channel capacity of the device.
6001, 6011 and up	-	RTCP traffic	Always adjacent to the RTP port number
6002, 6012 and up	-	T.38 traffic	Always adjacent to the RTCP port number
5060	5060	SIP	Configurable (LocalSIPPort [UDP], TCPLocalSIPPort [TCP]).
5061	5061	SIP over TLS (SIPS)	Configurable (TLSLocalSIPPort)
(random) > 32767	514	Syslog	Disabled by default (EnableSyslog).
(random) > 32767	-	Syslog ICMP	Disabled by default (EnableSyslog).
(random) > 32767	-	ARP listener	
(random) > 32767	162	SNMP Traps	Can be disabled (DisableSNMP)
(random) > 32767	-	DNS client	

14.7 Recommended Practices

To improve network security, the following guidelines are recommended when configuring the Mediant 1000:

- Set the password of the primary web user account (refer to 5.6.7.1 on page 170) to a unique, hard-to-hack string. Do not use the same password for several devices as a single compromise may lead to others. Keep this password safe at all times and change it frequently.
- If possible, use a RADIUS server for authentication. RADIUS allows you to set different passwords for different users of the Mediant 1000, with centralized management of the password database. Both Web and Telnet interfaces support RADIUS authentication (refer to Section 14.4 on page 357).
- If the number of users that access the Web and Telnet interfaces is limited, you can use the ‘Web and Telnet Access List’ to define up to ten IP addresses that are permitted to access these interfaces. Access from an undefined IP address is denied (refer to Section 5.6.7.2 on page 172).
- Use IPSec to secure traffic to all management and control hosts. Since IPSec encrypts all traffic, hackers cannot capture sensitive data transmitted on the network, and malicious intrusions are severely limited.
- Use HTTPS when accessing the Web interface. Set HTTPSOnly to 1 to allow only HTTPS traffic (and block port 80). If you don't need the Web interface, disable the Web server (DisableWebTask).
- If you use Telnet, do not use the default port (23). Use SSL mode to protect Telnet traffic from network sniffing.
- If you use SNMP, do not leave the community strings at their default values as they can be easily guessed by hackers (refer to Section 16.7.1 on page 377).
- Use a firewall to protect your VoIP network from external attacks. Network robustness may be compromised if the network is exposed to Denial of Service (DoS) attacks. DoS attacks are mitigated by Stateful firewalls. Do not allow unauthorized traffic to reach the Mediant 1000.

14.8 Legal Notice

By default, the Mediant 1000 supports export-grade (40-bit and 56-bit) encryption due to US government restrictions on the export of security technologies. To enable 128-bit and 256-bit encryption on your device, contact your AudioCodes representative.

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/)

This product includes cryptographic software written by Eric Young' (mailto:mail to: eay@cryptsoft.com).

Reader's Notes

15 Diagnostics

Several diagnostic tools are provided, enabling you to identify correct functioning of the Mediant 1000, or an error condition with a probable cause and a solution or workaround.

- Front and rear-panel LEDs on the Mediant 1000. The location and functionality of the Mediant 1000 front panel LEDs is shown in Section 2.1.6 on page 33.
- Self-Testing on hardware initialization (refer to Section 15.1 below).
- Line testing (refer to Section 15.2 on page 366).
- Error / notification messages via the following interfaces:
    - Syslog: Log messages can be viewed using an external Syslog server (refer to Section 15.3 on page 366), or in the 'Message Log' screen in the Embedded Web Server (refer to Section 5.7.2 on page 189). Note that the 'Message Log' screen is not recommended for prolong debugging.
    - RS-232 terminal (for establishing a serial communications link with the Mediant 1000, refer to Section 12.2 on page 319).

15.1 Self-Testing

The Mediant 1000 features two self-testing modes used to identify faulty hardware components:

- **Rapid:** The Rapid test is performed every time the Mediant 1000 starts up. It is executed each time the Mediant 1000 completes its initialization process. This is a short test phase in which the only error detected and reported is failure in initializing hardware components. If an error is detected, an error message is sent to the Syslog.
- **Detailed:** Used in addition to the Rapid test mode. The test is performed on startup, when initialization of the Mediant 1000 is completed and if the parameter EnableDiagnostics is set to 1 or 2. In this mode, the Mediant 1000 tests its DSPs, RAM and flash memory. When EnableDiagnostics is set to 1, flash is tested thoroughly, when EnableDiagnostics is set to 2, flash is only partially tested. While the Detailed test is running, the Ready and Fail LEDs are lit.
If an error is detected, an error message is sent to the Syslog.

Warning: To continue regular operation, the Detailed test must be disabled. Set the parameter EnableDiagnostics to 0, and then reset the Mediant 1000.

15.2 Mediant 1000 Line Testing

The Mediant 1000 features a mechanism that performs tests on the telephone lines connected to FXS and FXO ports. These tests provide various line measurements. Line testing is executed via SNMP only (using the acAnalogFxoLineTestTable SNMP table for FXO and the acAnalogFxsLineTestTable SNMP table for FXS).

Note: The line testing mechanism must only be used for monitoring and never when there are calls in progress.

The following line tests are available on FXS modules:

- Hardware revision number
- Temperature (above or below limit, only if a thermometer is installed)
- Hook state
- Coefficients checksum
- Message waiting indication status
- Ring state
- Reversal polarity state
- Line current (only on port 0)
- Line voltage between TIP and RING (only on port 0)
- 3.3 V reading (only on port 0)
- Ring voltage (only on port 0)
- Long line current (only on port 0)

The following line tests are available on FXO modules:

- Hardware revision number
- Hook state
- Reversal polarity state

15.3 Syslog Support

Syslog protocol is an event notification protocol that enables a machine to send event notification messages across IP networks to event message collectors -also known as Syslog servers. Syslog protocol is defined in the IETF RFC 3164 standard.

Since each process, application and operating system was written independently, there is little uniformity to Syslog messages. For this reason, no assumption is made on the contents of the messages other than the minimum requirements of its priority.

Syslog uses UDP as its underlying transport layer mechanism. By default, UDP port 514 is assigned to Syslog, but this can be changed (using the SyslogServerPort parameter).

The Syslog message is transmitted as an ASCII (American Standard Code for Information Interchange) message. The message starts with a leading ‘<’ ('less-than' character), followed by a number, which is followed by a ‘>’ ('greater-than' character). This is optionally followed by a single ASCII space.

The number described above is known as the Priority and represents both the Facility and Severity as described below. The Priority number consists of one, two, or three decimal integers.

For example:

```
<37> Oct 11 16:00:15 mymachine su: 'su root' failed for lonvick on
/dev/pts/8
```

15.3.1 Syslog Servers

Users can use the provided Syslog server (ACSyslog) or other third-party Syslog servers. Examples of Syslog servers available as shareware on the Internet:

- Kiwi Enterprises: http://www.kiwisyslog.com/
- The US CMS Server: http://uscms.fnal.gov/hanlon/uscms_server/
- TriAction Software: http://www.triaction.nl/Products/SyslogDaemon.asp
- Netal SL4NT 2.1 Syslog Daemon: http://www.netal.com/

A typical Syslog server application enables filtering of the messages according to priority, IP sender address, time, date, etc.

15.3.2 Operation

The Syslog client, embedded in the Mediant 1000, sends error reports/events generated by the Mediant 1000 unit application to a Syslog server, using IP/UDP protocol.

➢ **To activate the Syslog client on the Mediant 1000, take these 5 steps:**

1. Set the parameter ‘EnableSyslog’ to 1 (refer to Table 5-37 on page 179).
2. Use the parameter ‘SyslogServerIP’ to define the IP address of the Syslog server you use (refer to Table 5-37 on page 179).
3. Use the parameter 'SyslogServerPort' to define the port number of the Syslog server (refer to Table 5-37 on page 179).
4. To determine the Syslog logging level use the parameter ‘GWDebugLevel’ (refer to Table 5-7 on page 85).
5. To enable the gateway to send log messages that report certain types of Web actions according to a pre-defined filter use the parameter ‘ActivityListToLog’ (described in Table 5-37 on page 179).

Reader's Notes

16 SNMP-Based Management

Simple Network Management Protocol (SNMP) is a standards-based network control protocol for managing elements in a network. The SNMP Manager (usually implemented by a Network Management System (NMS) or an Element Management System (EMS)) connects to an SNMP Agent (embedded on a remote Network Element (NE)) to perform network element Operation, Administration and Maintenance (OAM).

Both the SNMP Manager and the NE refer to the same database to retrieve information or configure parameters. This database is referred to as the Management Information Base (MIB), and is a set of statistical and control values. Apart from the standard MIBs documented in IETF RFCs, SNMP additionally enables the use of private MIBs, containing a non-standard information set (specific functionality provided by the NE).

Directives, issued by the SNMP Manager to an SNMP Agent, consist of the identifiers of SNMP variables (referred to as MIB object identifiers or MIB variables) along with instructions to either get the value for that identifier, or set the identifier to a new value (configuration). The SNMP Agent can also send unsolicited events towards the EMS, called SNMP traps.

The definitions of MIB variables supported by a particular agent are incorporated in descriptor files, written in Abstract Syntax Notation (ASN.1) format, made available to EMS client programs so that they can become aware of MIB variables and their use.

The device contains an embedded SNMP Agent supporting both general network MIBs (such as the IP MIB), VoP-specific MIBs (such as RTP) and AudioCodes' proprietary MIBs (acBoard, acGateway, acAlarm and other MIBs), enabling a deeper probe into the inter-working of the device. All supported MIB files are supplied to customers as part of the release.

16.1 SNMP Standards and Objects

16.1.1 SNMP Message Standard

Four types of SNMP messages are defined:

- **Get:** A request that returns the value of a named object.
- **Get-Next:** A request that returns the next name (and value) of the ‘next’ object supported by a network device given a valid SNMP name.
- **Set:** A request that sets a named object to a specific value.
- **Trap:** A message generated asynchronously by network devices. It is an unsolicited message from an agent to the manager.

Each of these message types fulfills a particular requirement of network managers:

- **Get request:** Specific values can be fetched via the ‘get’ request to determine the performance and state of the device. Typically, many different values and parameters can be determined via SNMP without the overhead associated with logging into the device, or establishing a TCP connection with the device.
- **Get-Next request:** Enables the SNMP standard network managers to ‘walk’ through all SNMP values of a device (via the ‘get-next’ request) to determine all names and values that an operant device supports.
- **Get-Bulk:** Extends the functionality of Get-Next by allowing multiple values to be returned for selected items in the request. This is accomplished by beginning with the first SNMP object to be fetched, fetching the next name with a "get-next", and repeating this operation.

- **Set Request:** The SNMP standard provides a method of effecting an action associated with a device (via the 'set' request) to accomplish activities such as disabling interfaces, disconnecting users, clearing registers, etc. This provides a way of configuring and controlling network devices via SNMP.
- **Trap message:** The SNMP standard furnishes a mechanism by which devices can 'reach out' to a Network Manager on their own (via a 'trap' message) to notify or alert the manager of a problem with the device. This typically requires each device on the network to be configured to issue SNMP traps to one or more network devices that are awaiting these traps.

The above message types are all encoded into messages referred to as Protocol Data Units (PDUs) that are interchanged between SNMP devices.

16.1.2 SNMP MIB Objects

The SNMP MIB is arranged in a tree-structured fashion, similar in many ways to a disk directory structure of files. The top level SNMP branch begins with the ISO 'internet' directory, which contains the following four main branches:

- **"mgmt" SNMP branch:** Contains the standard SNMP objects usually supported (at least in part) by all network devices.
- **"private" SNMP branch:** Contains those 'extended' SNMP objects defined by network equipment vendors.
- **"experimental" and "directory" SNMP branches:** Also defined within the 'internet' root directory, these branches are usually devoid of any meaningful data or objects.

The 'tree' structure described above is an integral part of the SNMP standard, though the most pertinent parts of the tree are the 'leaf' objects of the tree that provide actual management data regarding the device. Generally, SNMP leaf objects can be partitioned into two similar, but slightly different types that reflect the organization of the tree structure:

- **Discrete MIB Objects:** Contain one precise piece of management data. These objects are often distinguished from "Table" items (below) by adding a ".0" (dot-zero) extension to their names. The operator must merely know the name of the object and no other information.
- **Table MIB Objects:** Contain multiple sections of management data. These objects are distinguished from 'Discrete' items (above) by requiring a "." (dot) extension to their names that uniquely distinguishes the particular value being referenced. The "." (dot) extension is the "instance" number of an SNMP object. For "Discrete" objects, this instance number is zero. For "Table" objects, this instance number is the index into the SNMP table. SNMP tables are special types of SNMP objects which allow parallel arrays of information to be supported. Tables are distinguished from scalar objects, so that tables can grow without bounds. For example, SNMP defines the "ifDescr" object (as a standard SNMP object) that indicates the text description of each interface supported by a particular device. Since network devices can be configured with more than one interface, this object can only be represented as an array.

By convention, SNMP objects are always grouped in an "Entry" directory, within an object with a "Table" suffix. (The ifDescr object described above resides in the "ifEntry" directory contained in the "ifTable" directory).

16.1.3 SNMP Extensibility Feature

One of the principal components of an SNMP manager is a MIB Compiler which allows new MIB objects to be added to the management system. When a MIB is compiled into an SNMP manager, the manager is made "aware" of new objects that are supported by agents on the network. The concept is similar to adding a new schema to a database.

Typically, when a MIB is compiled into the system, the manager creates new folders or directories that correspond to the objects. These folders or directories can typically be viewed with a MIB Browser, which is a traditional SNMP management tool incorporated into virtually all Network Management Systems.

The act of compiling the MIB allows the manager to know about the special objects supported by the agent and access these objects as part of the standard object set.

16.2 Carrier Grade Alarm System

The basic alarm system has been extended to a carrier-grade alarm system. A carrier-grade alarm system provides a reliable alarm reporting mechanism that takes into account EMS outages, network outages, and transport mechanism such as SNMP over UDP.

A carrier-grade alarm system is characterized by the following:

- The device has a mechanism that allows a manager to determine which alarms are currently active in the device. That is, the device maintains an active alarm table.
- The device has a mechanism to allow a manager to detect lost alarm raise and clear notifications [sequence number in trap, current sequence number MIB object].
- The device has a mechanism to allow a manager to recover lost alarm raise and clear notifications [maintains a log history].
- The device sends a cold start trap to indicate that it is starting. This allows the EMS to synchronize its view of the device's active alarms.

When the SNMP alarm traps are sent, the carrier-grade alarm system does not add or delete alarm traps as part of the feature. This system provides the mechanism for viewing of history and current active alarm information.

16.2.1 Active Alarm Table

The device maintains an active alarm table to allow a manager to determine which alarms are currently active in the device. Two views of the active alarm table are supported by the agent:

- acActiveAlarmTable in the enterprise acAlarm
- alarmActiveTable and alarmActiveVariableTable in the IETF standard ALARM-MIB (rooted in the AC tree)

The acActiveAlarmTable is a simple, one-row per alarm table that is easy to view with a MIB browser.

The ALARM-MIB is currently a draft standard and therefore has no OID assigned to it. In the current software release, the MIB is rooted in the experimental MIB subtree. In a future release, after the MIB has been ratified and an OID assigned, it is to move to the official OID.

16.2.2 Alarm History

The device maintains a history of alarms that have been raised and traps that have been cleared to allow a manager to recover any lost, raised or cleared traps. Two views of the alarm history table are supported by the agent:

- acAlarmHistoryTable in the enterprise acAlarm
- nlmLogTable and nlmLogVariableTable in the standard NOTIFICATION-LOG-MIB

As with the acActiveAlarmTable, the acAlarmHistoryTable is a simple, one-row-per-alarm table that is easy to view with a MIB browser.

16.3 Cold Start Trap

Mediant 1000 technology supports a cold start trap to indicate that the device is starting. This allows the manager to synchronize its view of the device's active alarms. Two different traps are sent at start-up:

- The standard coldStart trap - iso(1).org(3).dod(6).internet(1). snmpV2(6). snmpModules(3). snmpMIB(1). snmpMIBObjects(1). snmpTraps(5). coldStart(1) - sent at system initialization.
- The enterprise acBoardEvBoardStarted which is generated at the end of system initialization. This is more of an ‘application-level’ cold start sent after the entire initializing process is complete and all the modules are ready.

16.4 Third-Party Performance Monitoring Measurements

Performance measurements are available for a third-party performance monitoring system (or EMS) through an SNMP interface. These measurements can be polled at scheduled intervals by an external poller or utility in a media server or other off-device system.

The device provides two types of performance measurements:

- **Gauges:** Gauges represent the current state of activities on the device. Gauges, unlike counters, can decrease in value, and like counters, can increase. The value of a gauge is the current value or a snapshot of the current activity on the device.
- **Counters:** Counters always increase in value and are cumulative. Counters, unlike gauges, never decrease in value unless the off-device system is reset. The counters are then zeroed.

Performance measurements are provided by several proprietary MIBs that are located under the acPerformance sub tree:

iso(1).org(3).dod(6).internet(1).private(4).enterprises(1).audioCodes(5003).acPerformance(10)

Two formats of performance monitoring MIBs are available:

- Old format (obsolete as of version 4.6):
  Each MIB is composed of a list of single MIB objects, each relates to a separate attribute within a gauge or a counter. All counters and gauges provide the current time value only.
  - **acPerfMediaGateway:** a generic-type of PM MIB that covers:
    - Control protocol
    - RTP stream
    - System packets statistics
  - **acPerfMediaServices:** media services devices specific performance MIB.
  - **acPerfH323SIPGateway:** holds statistics on Tel to IP and vice versa.
- New format:
  The following MIBs feature an identical structure. Each includes two major sub-trees.
  - Configuration sub tree – enables configuration of general attributes of the MIB and specific attributes of the monitored objects.
  - Data sub tree

  The monitoring results are presented in tables. Each table includes one or two indices. When there are two indices, the first index is a sub-set in the table (e.g., trunk number) and the second (or a single where there is only one) index represents the interval number (present - 0, previous - 1 and the one before - 2).

  The MIBs include:

  - **acPMMedia:** for media (voice) related monitoring (e.g., RTP, DSP’s).
  - **acPMControl:** for Control-Protocol related monitoring (e.g., connections, commands).

- **acPMPSTN:** for PSTN related monitoring such as channel use, trunk utilization. Note that the acPMTrunkUtilizationTable is not supported.
- **acPMSystem:** for general (system related) monitoring.
- The log trap, acPerformanceMonitoringThresholdCrossing (non-alarm) is sent every time the threshold of a Performance Monitored object is crossed. The severity field is "indeterminate" when the crossing is above the threshold and "cleared" when it falls bellow the threshold. The "source" varbind in the trap indicates the object for which the threshold is being crossed.

16.4.1 Total Counters

The counter's attribute 'total' accumulates counter values since the board's most recent restart. The user can reset the total's value by setting the Reset-Total object.

Each MIB module has its own Reset Total object, as follows:

- **PM-Analog:** acPMAnalogConfigurationResetTotalCounters
- **PM-ATM:** acPMAtmConfigurationResetTotalCounters (not applicable to this release)
- **PM-Control:** acPMControlConfigurationResetTotalCounters
- **PM-Media:** acPMMediaConfigurationResetTotalCounters
- **PM-PSTN:** acPMPSTNConfigurationResetTotalCounters
- **PM-System:** acPMSystemConfigurationResetTotalCounters

16.5 TrunkPack-VoP Series Supported MIBs

The Mediant 1000 contains an embedded SNMP Agent supporting the following MIBs:

- **Standard MIB (MIB-2):** The various SNMP values in the standard MIB are defined in RFC 1213. The standard MIB includes various objects to measure and monitor IP activity, TCP activity, UDP activity, IP routes, TCP connections, interfaces and general system indicators.
- **RTP MIB:** The RTP MIB is supported in conformance with the IETF RFC 2959. It contains objects relevant to the RTP streams generated and terminated by the device and to RTCP information related to these streams.
- **Trunk MIB:** The Trunk MIB contains objects relevant to E1/T1 Trunk interfaces.
- **NOTIFICATION-LOG-MIB:** This standard MIB (RFC 3014 - iso.org.dod.internet.mgmt.mib-2) is supported as part of our implementation of carrier grade alarms.
- **ALARM-MIB:** This is an IETF MIB (RFC 3877) also supported as part of our implementation of carrier grade alarms. This MIB is a new standard and is therefore under the audioCodes.acExperimental branch.
- **SNMP-TARGET-MIB:** According to RFC 2273. It allows for the configuration of trap destinations and trusted managers.
- **SNMP MIB:** This MIB (RFC 3418) allows support of the coldStart and authenticationFailure traps.
- **SNMP Framework MIB:** (RFC 3411).
- **SNMP Usm MIB:** this MIB (RFC 3414) implements the user-based Security Model.
- **SNMP Vacm MIB:** This MIB (RFC 3415) implements the view-based Access Control Model.
- **SNMP Community MIB:** This MIB (RFC 3584). implements community string management.

- **RTCP-XR:** This MIB (RFC) implements the following partial support:
  - The rtcpXrCallQualityTable is fully supported.
  - In the rtcpXrHistoryTable, support of the RCQ objects is provided only with no more than 3 intervals, 15 minutes long each.
  - Supports the rtcpXrVoipThresholdViolation trap.
- **ds1 MIB:** support for the following:
  - dsx1ConfigTable - partial supports following objects have SET and GET applied:
    - dsx1LineCoding
    - dsx1LoopbackConfig
    - dsx1LineStatusChangeTrapEnable
    - dsx1CircuitIdentifier

  All other objects in this table support GET only:
  - dsx1CurrentTable
  - dsx1IntervalTable
  - dsx1TotalTable
  - dsx1LineStatusChange trap
- **ipForward MIB:** (RFC 2096) fully supported

In addition to the standard MIBs, the complete product series contains proprietary MIBs:

- **AC-TYPES MIB:** lists the known types defined by the complete product series. This is referred to by the sysObjectID object in the MIB-II.

Note: The acBoard MIB is still supported but is being replaced by five newer proprietary MIBs. The only relevant section in this MIB is the trap sub-tree acTrap.

As noted above, five new MIBs cover the device's general parameters. Each contains a Configuration subtree for configuring related parameters. In some, there also are Status and Action subtrees.

The five MIBs are:

- **AC-CONTROL-MIB**
- **AC-MEDIA-MIB**
- **AC-PSTN-MIB**
- **AC-SYSTEM-MIB**
- **AC-SS7-MIB**
- **acGateway MIB:** This proprietary MIB contains objects related to configuration of the device when applied as a SIP or H.323 media gateway only. This MIB complements the other proprietary MIBs.

  The acGateway MIB has the following groups:
  - Common - for parameters common to both SIP and H.323
  - SIP - for SIP parameters only
  - H.323 - for H.323 parameters only
- **acAtm:** This proprietary MIB contains objects related to configuration and status of the device when applied as an ATM media gateway only. This MIB complements the other proprietary MIBs.
  The acAtm MIB has the following groups:
  - **acAtmConfiguration:** for configuring ATM related parameters
  - **acAtmStatus:** for the status of ATM connections

- **acAlarm:** This is AudioCodes' proprietary carrier-grade alarm MIB. It is a simpler implementation of the notificationLogMIB and the IETF suggested alarmMIB (both also supported in all AudioCodes' devices).
  The acAlarm MIB has the following groups:
  - **ActiveAlarm:** straightforward (single-indexed) table, listing all currently active alarms, together with their bindings (the alarm bindings are defined in acAlarm. acAlarmVarbinds and also in acBoard.acTrap. acBoardTrapDefinitions. oid_1_3_6_1_4_1_5003_9_10_1_21_2_0).
  - **acAlarmHistory:** straightforward (single-indexed) table, listing all recently raised alarms together with their bindings (the alarm bindings are defined in acAlarm. acAlarmVarbinds and also in acBoard.acTrap. acBoardTrapDefinitions. oid_1_3_6_1_4_1_5003_9_10_1_21_2_0).

The table size can be altered via notificationLogMIB.notificationLogMIBObjects.nlmConfig.nlmConfigGlobalEntryLimit or notificationLogMIB.notificationLogMIBObjects.nlmConfig.nlmConfigLogTable.nlm ConfigLogEntry.nlmConfigLogEntryLimit.

The table size can be any value between 50 and 1000 (default is 500).

Note 1: The following are special notes pertaining to MIBs:

- A detailed explanation of each parameter can be viewed in an SNMP browser in the "MIB Description" field.
- Not all groups in the MIB are functional. Refer to version release notes.
- Certain parameters are non-functional. Their MIB status is marked "obsolete".
- When a parameter is set to a new value via SNMP, the change may affect device functionality immediately or may require that the device be soft reset for the change to take effect. This depends on the parameter type.

Note 2: The current (updated) device configuration parameters are programmed into the device provided that the user does not load an *ini* file to the device after reset. Loading an *ini* file after reset overrides the updated parameters.

16.6 Traps

Note: All traps are sent from the SNMP port (default 161). This is part of the NAT traversal solution.

Full proprietary trap definitions and trap Varbinds are found in the acBoard MIB and acAlarm MIB. Table 16-1 lists the supported proprietary traps. For detailed information on these traps, refer to Appendix I on page 451.

Table 16-1: Proprietary Traps Description (continues on pages 376 to 377)

Trap	Description
acBoardFatalError	Sent whenever a fatal device error occurs.
acBoardConfigurationError	Sent when a device’s settings are illegal. The trap contains a message describing the illegality of the setting. Note: Not applicable to IPM-260).
acBoardTemperatureAlarm	Sent when a board exceeds its temperature limits.
acBoardEvResettingBoard	Sent after the device is reset.
acBoardEvBoardStarted	Sent after the device is successfully restored and initialized following reset.
acgwAdminStateChange	Sent when Graceful Shutdown commences and ends.
acOperationalStateChange	Sent if the operational state of the node changes to disabled. Cleared when the operational state of the node changes to enabled.
acBoardCallResourcesAlarm	Sent when no free channels are available.
acBoardControllerFailureAlarm	Sent when the Gatekeeper/Proxy is not found or registration failed. Internal routing table can be used for routing.
acFeatureKeyError	Intended to relay Feature Key errors etc. (will be supported in the next applicable release).
acBoardOverloadAlarm	Sent when an overload in one or more of the system's components occurs.
acActiveAlarmTableOverflow	Sent to indicate that an active alarm could not be entered into the Active Alarm table because the table was full.
acKeepAlive	Part of the NAT traversal mechanism. If the STUN application detects a NAT, this trap is sent on regular time laps - 9/10 of the acSysSTUNBindingLifeTime object. The AdditionalInfo1 varbind has the MAC address of the gateway.
acNATTraversalAlarm	Sent when the NAT is placed in front of a gateway that is identified as a symmetric NAT. It is cleared when a non-symmetric NAT or no NAT replaces the symmetric one.
acEnhancedBITStatus	This trap is used to indicate the status of the Built In Test (BIT). The information in the trap contains board hardware elements being tested and their status. The information is presented in the additional info fields.
acBoardEthernetLinkAlarm	Sent when an Ethernet link(s) is down.
acPerformanceMonitoringThresholdCrossing	Sent every time the threshold of a Performance Monitored object is crossed. The severity field is 'indeterminate' when the crossing is above the threshold, and 'cleared' when it goes back under the threshold. The 'source' varbind in the trap indicates the object for which the threshold is being crossed.

Table 16-1: Proprietary Traps Description (continues on pages 376 to 377)

Trap	Description
acHTTPDownloadResult	Sent at the success or failure of HTTP download.
acDChannelStatus	Non-alarm trap sent at the establishment, re-establishment, or release of LAPD link with its peer connection. The trap is sent with one of the following textual descriptions: D-channel synchronized D-channel not-synchronized

In addition to the listed traps, the device also supports the following standard traps:

- dsx1LineStatusChange
- dsx3LineStatusChange
- coldStart
- authenticationFailure

16.7 SNMP Interface Details

This section describes details of the SNMP interface that is required when developing an Element Manager (EM) for any of the TrunkPack-VoP Series products, or to manage a device with a MIB browser.

The gateway offers the following SNMP security features:

- SNMPv2c community strings
- SNMPv3 User-based Security Model (USM) users
- SNMP encoded over IPSec (refer to Section 14.1 on page 343)
- Combinations of the above

Currently, both SNMP and *ini* file commands and downloads are not encrypted. For *ini* file encoding, refer to Section 6.1 on page 213.

16.7.1 SNMP Community Names

By default, the device uses a single, read-only community string of ‘public’ and a single read-write community string of ‘private’.

The following community strings can be defined:

- Up to five read-only community strings
- Up to five read-write community strings
- A single trap community string

Each community string must be associated with one of the following predefined SNMP groups:

Table 16-2: SNMP Predefined Groups

Group	Get Access	Set Access	Can Send Traps
ReadGroup	Yes	No	Yes
ReadWriteGroup	Yes	Yes	Yes
TrapGroup	No	No	Yes

16.7.1.1 Configuration of Community Strings via the Web

For detailed information on configuration the community strings via the Embedded Web Server, refer to Section 5.6.8.2 on page 181.

16.7.1.2 Configuration of Community Strings via the *ini* File

The following *ini* file parameters are used to configure community strings:

- SNMPReadOnlyCommunityString_<x> = '#######'
- SNMPReadWriteCommunityString_<x> = '#######'

Where <x> is a number from 0 to 4. The '#' character represents any alphanumeric character. The maximum length of the string is 20 characters.

16.7.1.3 Configuration of Community Strings via SNMP

To configure read-only and read-write community strings, the EM must use the snmpCommunityMIB. To configure the trap community string, the EM must also use the snmpVacmMIB and the snmpTargetMIB.

➢ To add a read-only community string (v2user), take these 2 steps:

1. Add a new row to the snmpCommunityTable with CommunityName v2user.
2. Add a row to the vacmSecurityToGroupTable for SecurityName v2user, GroupName ReadGroup and SecurityModel snmpv2c.

➢ To delete the read-only community string (v2user), take these 3 steps:

1. If v2user is being used as the trap community string, follow the procedure for changing the trap community string. (See below.)
2. Delete the snmpCommunityTable row with CommunityName v2user.
3. Delete the vacmSecurityToGroupTable row for SecurityName v2user, GroupName ReadGroup and SecurityModel snmpv2c.

➢ To add a read-write community string (v2admin), take these 2 steps:

1. Add a new row to the snmpCommunityTable with CommunityName v2admin.
2. Add a row to the vacmSecurityToGroupTable for SecurityName v2admin, GroupName ReadWriteGroup and SecurityModel snmpv2c.

➢ To delete the read-write community string (v2admin), take these 2 steps:

1. If v2admin is being used as the trap community string, follow the procedure for changing the trap community string. (Refer to the procedure below.)
2. Delete the snmpCommunityTable row with a CommunityName of v2admin and GroupName of ReadWriteGroup.

➢ To change the only read-write community string from v2admin to v2mgr, take these 4 steps:

1. Follow the procedure above to add a read-write community string to a row for v2mgr.
2. Set up the EM so that subsequent ‘set’ requests use the new community string, v2mgr.
3. If v2admin is being used as the trap community string, follow the procedure to change the trap community string (see below).
4. Follow the procedure above to delete a read-write community name in the row for v2admin.

➢ To change the trap community string, take these 3 steps:

1. Add a row to the vacmSecurityToGroupTable with these values:
   - SecurityModel = 2
   - SecurityName = the new trap community string
   - GroupName = TrapGroup, ReadGroup or ReadWriteGroup
   - The SecurityModel and SecurityName objects are row indices
2. Modify the SecurityName field in the sole row of the snmpTargetParamsTable.
3. Remove the row from the vacmSecurityToGroupTable with SecurityName = the old trap community string.

Notes:

- The procedure above assumes that a row already exists in the snmpCommunityTable for the new trap community string. The trap community string can be part of the TrapGroup, ReadGroup, or ReadWriteGroup. If the trap community string is used solely for sending traps (recommended), then it should be made part of the TrapGroup.
- You must add GroupName and RowStatus on the same set.

16.7.2 SNMP v3 USM Users

You can define up to 10 User-based Security Model (USM) users (USM users are referred to as “v3 users”). Each v3 user can be associated with an authentication type (none, MD5, or SHA-1) and a privacy type (none, DES, 3DES, or AES).

Table 16-3: SNMP v3 Security Levels

Security Level	Authentication	Privacy
noAuthNoPriv(1)	None	None
authNoPriv(2)	MD5 or SHA-1	None
authPriv(3)	MD5 or SHA-1	DES, 3DES, AES128, AES192, or AES256

Each SNMP v3 user must be associated with one of the predefined groups listed in the following table:

Table 16-4: SNMP v3 Predefined Groups

Group	Get Access	Set Access	Send Traps	Security Level
ReadGroup1	Yes	No	Yes	noAuthNoPriv(1)
ReadWriteGroup1	Yes	Yes	Yes	noAuthNoPriv(1)
TrapGroup1	No	No	Yes	noAuthNoPriv(1)
ReadGroup2	Yes	No	Yes	authNoPriv(2)
ReadWriteGroup2	Yes	Yes	Yes	authNoPriv(2)
TrapGroup2	No	No	Yes	authNoPriv(2)
ReadGroup3	Yes	No	Yes	authPriv(3)
ReadWriteGroup3	Yes	Yes	Yes	authPriv(3)
TrapGroup3	No	No	Yes	authPriv(3)

16.7.2.1 Configuring SNMP v3 users via the *ini* File

Use the SNMPUsers *ini* table to add, modify, and delete SNMPv3 users. For a description of the SNMPUsers table *ini* file parameters, refer to Section 6.5.6 on page 229.

Note: The SNMPUsers *ini* table is a hidden parameter. Therefore, when you perform a “Get ini File” operation using the Web interface, the table will not be included in the generated file.

You can enter keys in the form of a text password or in the form of a localized key in hex format. If using a text password, then it should be at least eight characters in length. Below is an example of a localized key format:

```
26:60:d8:7d:0d:4a:d6:8c:02:73:dd:22:96:a2:69:df
```

The following example configuration creates three SNMPv3 USM users:

```
[ SNMPUsers ]
FORMAT SNMPUsers_Index = SNMPUsers_Username, SNMPUsers_AuthProtocol,
SNMPUsers_PrivProtocol, SNMPUsers_AuthKey, SNMPUsers_PrivKey, SNMPUsers_Group;
SNMPUsers 0 = v3user, 0, 0, -, -, 0;
SNMPUsers 1 = v3admin1, 1, 0, myauthkey, -, 1;
SNMPUsers 2 = v3admin2, 2, 1, myauthkey, myprivkey, 1;
[ \SNMPUsers ]
```

The example above creates the following three v3 users:

- The user "v3user" is defined for a security level of noAuthNoPriv(1) and is associated with ReadGroup1.
- The user "v3admin1" is defined for a security level of authNoPriv(2) with authentication protocol MD5. The authentication text password is “myauthkey” and the user will be associated with ReadWriteGroup2.
- The user "v3admin2" is defined for a security level of authPriv(3) with authentication protocol SHA-1 and privacy protocol DES. The authentication text password is “myauthkey”, the privacy text password is “myprivkey”, and the user will be associated with ReadWriteGroup3.

16.7.2.2 Configuring SNMP v3 Users via SNMP

To configure SNMP v3 users, the EM must use the standard snmpUsmMIB and the snmpVacmMIB.

➢ To add a read-only, noAuthNoPriv SNMPv3 user (v3user), take these 3 steps:

1. Clone the row with the same security level. After the clone step, the status of the row is notReady(3).
2. Activate the row (i.e., set the row status to active(1)).
3. Add a row to the vacmSecurityToGroupTable for SecurityName v3user, GroupName ReadGroup1, and SecurityModel usm(3).

Note: A row with the same security level (noAuthNoPriv) must already exist in the usmUserTable. (See the usmUserTable for details).

➢ To delete the read-only, noAuthNoPriv SNMPv3 user (v3user), take these 3 steps:

1. If v3 user is associated with a trap destination, follow the procedure for associating a different user to that trap destination. (See below.)
2. Delete the vacmSecurityToGroupTable row for SecurityName v3user, GroupName ReadGroup1, and SecurityModel usm.
3. Delete the row in the usmUserTable for v3user.

➢ To add a read-write, authPriv SNMPv3 user (v3user), take these 4 steps:

1. Clone the row with the same security level.
2. Change the authentication key and privacy key.
3. Activate the row. That is, set the row status to active(1).
4. Add a row to the vacmSecurityToGroupTable for SecurityName v3admin1, GroupName ReadWriteGroup3, and SecurityModel usm(3).

Note: A row with the same security level (authPriv) must already exist in the usmUserTable (see the usmUserTable for details).

➢ To delete the read-write, authPriv SNMPv3 user (v3admin1), take these 3 steps:

1. If v3admin1 is associated with a trap destination, follow the procedure for associating a different user to that trap destination. (See below.)
2. Delete the vacmSecurityToGroupTable row for SecurityName v3admin1, GroupName ReadWriteGroup1, and SecurityModel usm.
3. Delete the row in the usmUserTable for v3admin1.

16.7.3 Trusted Managers

By default, the agent accepts ‘get’ and ‘set’ requests from any IP address, as long as the correct community string is used in the request. Security can be enhanced via the use of Trusted Managers. A Trusted Manager is an IP address from which the SNMP Agent accepts and processes ‘get’ and ‘set’ requests. An EM can be used to configure up to five Trusted Managers.

Note: If Trusted Managers are defined, all community strings work from all Trusted Managers. That is, there is no way to associate a community string with particular trusted managers.

The concept of trusted managers is considered to be a weak form of security and is therefore, not a required part of SNMPv3 security, which uses authentication and privacy. However, the board’s SNMP agent applies the trusted manager concept as follows:

- There is no way to configure trusted managers for only a SNMPv3 user. An SNMPv2c community string must be defined.
- If specific IPs are configured as trusted managers (via the community table), then only SNMPv3 users on those trusted managers are given access to the agent’s MIB objects.

16.7.3.1 Configuration of Trusted Managers via *ini* File

To set the Trusted Mangers table from start-up, write the following in the *ini* file:

SNMPTRUSTEDMGR_X = D.D.D.D

where X is any integer between 0 and 4 (0 sets the first table entry, 1 sets the second, and so on), and D is an integer between 0 and 255.

16.7.3.2 Configuration of Trusted Managers via SNMP

To configure Trusted Managers, the EM must use the SNMP-COMMUNITY-MIB, the snmpTargetMIB and the snmpTargetMIB.

The procedure below assumes that there is at least one configured read-write community, are currently no Trusted Managers, and the TransportTag for columns for all snmpCommunityTable rows are currently empty.

➢ To add the first Trusted Manager, take these 3 steps:

1. Add a row to the snmpTargetAddrTable with these values:
   - Name=mgr0
   - TagList=MGR
   - Params=v2cparams
2. Add a row to the snmpTargetAddrExtTable table with these values:
   - Name=mgr0
   - snmpTargetAddrTMask=255.255.255.255:0.

   The agent does not allow creation of a row in this table unless a corresponding row exists in the snmpTargetAddrTable.
3. Set the value of the TransportTag field on each non-TrapGroup row in the snmpCommunityTable to MGR.

The following procedure assumes that there is at least one configured read-write community, are currently one or more Trusted Managers, and the TransportTag for columns for all rows in the snmpCommunityTable are currently set to MGR. This procedure must be done from one of the existing Trusted Managers.

➢ To add a subsequent Trusted Manager, take these 2 steps:

1. Add a row to the snmpTargetAddrTable with these values:
   - Name=mgrN
   - TagList=MGR
   - Params=v2cparams

   Where N is an unused number between 0 and 4.
2. Add a row to the snmpTargetAddrExtTable table with these values:
   - Name=mgrN
   - snmpTargetAddrTMask=255.255.255.255:0

An alternative to the above procedure is to set the snmpTargetAddrTMask column while you are creating other rows in the table.

The procedure below assumes that there is at least one configured read-write community, are currently two or more Trusted Managers, and the taglist for columns for all rows in the snmpCommunityTable are currently set to MGR. This procedure must be done from one of the existing trusted managers, but not the one that is being deleted.

➢ To delete a Trusted Manager (not the final one), take this step:

- Remove the appropriate row from the snmpTargetAddrTable.

The change takes affect immediately. The deleted trusted manager cannot access the board. The agent automatically removes the row in the snmpTargetAddrExtTable.

The procedure below assumes that there is at least one configured read-write community, currently only one Trusted Manager, and the taglist for columns for all rows in the snmpCommunityTable are currently set to MGR. This procedure must be done from thefinal Trusted Manager.

➢ To delete the final Trusted Manager, take these 2 steps:

1. Set the value of the TransportTag field on each row in the snmpCommunityTable to the empty string.
2. Remove the appropriate row from the snmpTargetAddrTable.

The change takes affect immediately. All managers can now access the board. The agent automatically removes the row in the snmpTargetAddrExtTable.

16.7.4 SNMP Ports

The SNMP Request Port is 161 and the Trap Port is 162. These ports can be changed by setting parameters in the device *ini* file. The parameter name is:

SNMPPort = <port_number>
Valid UDP port number; default = 161

This parameter specifies the port number for SNMP requests and responses. Usually, it should not be specified. Use the default.

16.7.5 Multiple SNMP Trap Destinations

An agent can send traps to up to five managers. For each manager, set the manager's IP address, receiving port number, and enable sending traps to that manager.

The user also has the option of associating a trap destination with a specific SNMPv3 USM user. Traps are then sent to that trap destination using the SNMPv3 format and the authentication and privacy protocol configured for that user..

To configure the trap managers table use:

- The Embedded Web Server (refer to Section 5.6.8.1 on page 180).
- The *ini* file (refer to Section 16.7.5.2 below).
- SNMP (refer to Section 16.7.5.3 on page 386).

16.7.5.1 Configuring Trap Manager via Host Name

One of the five available SNMP managers can be defined using a FQDN. In the current version, this option can only be configured via the *ini* file (SNMPTrapManagerHostName).

The gateway tries to resolve the host name at start up. Once the name is resolved (IP is found), the resolved IP address replaces the last entry in the trap manager table (defined by the parameter SNMPManagerTableIP_x) and the last trap manager entry of snmpTargetAddrTable in the snmpTargetMIB. The port is 162 (unless specified otherwise), the row is marked as 'used' and the sending is 'enabled'.

When using 'host name' resolution, any changes made by the user to this row in either MIBs are overwritten by the gateway when a resolving is redone (once an hour).

Note that several traps may be lost until the resolving is complete.

16.7.5.2 Configuring Trap Managers via the ini File

In the Mediant 1000 *ini* file, the parameters below can be set to enable or disable the sending of SNMP traps. Multiple trap destinations can be supported on the device by setting multiple trap destinations in the *ini* file.

- **SNMPManagerTrapSendingEnable_<x>:** indicates whether or not traps are to be sent to the specified SNMP trap manager. A value of '1' means that it is enabled, while a value of '0' means disabled. <x> represents a number 0, 1, 2 which is the array element index. Currently, up to five SNMP trap managers can be supported.
- **SNMPManagerTrapUser_<x>**: indicates to send an SNMPv2 trap using the trap user community string configured with the SNMPTrapCommunityString parameter. The user may instead specify an SNMPv3 user name.

Figure 16-1 presents an example of entries in a device's *ini* file regarding SNMP. The device can be configured to send to multiple trap destinations. The lines in the file below are commented with the ';' at the beginning of the line. All of the lines below are commented since the first line character is a semi-colon.

Figure 16-1: Example of Entries in a Device *ini* file Regarding SNMP

```
; SNMP trap destinations
; The board maintains a table of trap destinations containing 5 ;rows. The rows
are numbered 0..4. Each block of 4 items below ;apply to a row in the table.

; To configure one of the rows, uncomment all 4 lines in that ;block. Supply an
IP address and if necessary, change the port ;number.
; To delete a trap destination, set ISUSED to 0.
; -change these entries as needed
;SNMPManagerTableIP_0=
;SNMPManagerTrapPort_0=162
;SNMPManagerIsUsed_0=1
;SNMPManagerTrapSendingEnable_0=1
;
;SNMPManagerTableIP_1=
;SNMPManagerTrapPort_1=162
;SNMPManagerIsUsed_1=1
;SNMPManagerTrapSendingEnable_1=1
;
;SNMPManagerTableIP_2=
;SNMPManagerTrapPort_2=162
;SNMPManagerIsUsed_2=1
;SNMPManagerTrapSendingEnable_2=1
;
;SNMPManagerTableIP_3=
;SNMPManagerTrapPort_3=162
;SNMPManagerIsUsed_3=1
;SNMPManagerTrapSendingEnable_3=1
;
;SNMPManagerTableIP_4=
;SNMPManagerTrapPort_4=162
;SNMPManagerIsUsed_4=1
;SNMPManagerTrapSendingEnable_4=1
```

To configure the trap manger host name, use the parameter SNMPTrapManagerHostName. For example:

```
;SNMPTrapManagerHostName = 'myMananger.corp.MyCompany.com'
```

Note: The same information configurable in the *ini* file can also be configured via the acBoardMIB.

16.7.5.3 Configuring Trap Managers via SNMP

The standard snmpTargetMIB interface is available for configuring trap managers.

Note: The acBoard MIB is planned to become obsolete. The only relevant section in this MIB is the trap sub tree acTrap.

➢ To add an SNMPv2 trap destination, take this step:

- Add a row to the snmpTargetAddrTable with these values:
  - Name=trapN (where N is an unused number between 0 and 4)
  - TagList=AC_TRAP
  - Params=v2cparams

  All changes to the trap destination configuration take effect immediately.

➢ To add an SNMPv3 trap destination, take these 2 steps:

1. Add a row to the snmpTargetAddrTable with these values:
   - Name=trapN, where N is an unused number between 0 and 4
   - TagList=AC_TRAP
   - Params=usm<user>, where <user> is the name of the SNMPv3 with which this user is associated
2. If a row does not already exist for this combination of user and SecurityLevel, add a row to the snmpTargetParamsTable with these values:
   - Name=usm<user>
   - MPModel=3(SNMPv3)
   - SecurityModel=3 (usm)
   - SecurityName=<user>
   - SecurityLevel=M, where M is either 1(noAuthNoPriv), 2(authNoPriv), or 3(authPriv)

All changes to the trap destination configuration take effect immediately.

➢ To delete a trap destination, take this step:

- Remove the appropriate row from the snmpTargetAddrTable.

If this is the last trap destination associated with this user and security level, you can also delete the appropriate row from the snmpTargetParamsTable.

You can change the IP address and/or port number of an existing trap destination. The same effect can be achieved by removing a row and adding a new row.

➢ To modify a trap destination, take this step:

- Modify the IP address and/or port number for the appropriate row in the snmpTargetAddrTable.

➢ To disable a trap destination, take this step:

- Change TagList on the appropriate row in the snmpTargetAddrTable to the empty string.

➢ To enable a trap destination, take this step:

- Change TagList on the appropriate row in the snmpTargetAddrTable to ‘AC_TRAP’.

16.8 SNMP Manager Backward Compatibility

With support for the Multi Manager Trapping feature, the older acSNMPManagerIP MIB object, which is synchronized with the first index in the snmpManagers MIB table is also supported. This is translated in two features:

- SET/GET to either of the two MIB objects is identical. In other words, as far as the SET/GET are concerned, OID 1.3.6.1.4.1.5003.9.10.1.1.2.7 is identical to OID 1.3.6.1.4.1.5003.9.10.1.1.2.21.1.1.3.
- When setting ANY IP to the acSNMPManagerIP (this is the older parameter, not the table parameter), two more parameters are SET to ENABLE. snmpManagerIsUsed.0 and snmpManagerTrapSendingEnable.0 are both set to 1.

16.9 Dual Module Interface

Dual module boards comprise of two modules (the first module is on the right side of the board when looking at it from the front). As the two modules reside in a single board it is important to differentiate between them. Differentiation is based on the modules’ serial numbers.

MIB object acSysIdSerialNumber always returns the serial number of the module on which the GET operation is performed.

MIB object acSysIdFirstSerialNumber always returns the serial number of the first module.

If the module on which the GET operation is performed is the second module, then the values of these two objects are different. If, on the other hand, the module is the first module, the values of these two objects are the same.

16.10 SNMP NAT Traversal

A NAT placed between the gateway and the element manager calls for traversal solutions:

- **Trap source port:** all traps are sent out from the SNMP port (default 161). A manager receiving these traps can use the binding information (in the UDP layer) to traverse the NAT back to the device.
  The trap destination address (port and IP) are as configured in the snmpTargetMIB.
- **acKeepAliveTrap:** this trap is designed to be a constant life signal from the device to the manager allowing the manager NAT traversal at all times. The acBoardTrapGlobalsAdditionalInfo1 varbind has the device's serial number.
  The destination port (i.e., the manager port for this trap) can be set to be different than the port to which all other traps are sent. To do this, use the acSysSNMPKeepAliveTrapPort object in the acSystem MIB or the *ini* file parameter KeepAliveTrapPort.
  The Trap is instigated in three ways:
  - Via an *ini* file parameter (SendKeepAliveTrap = 1). This ensures that the trap is continuously sent. The frequency is set via the 9/10 of the acSysSTUNBindingLifeTime object.
  - After the STUN client has discovered a NAT (any NAT).
  - If the STUN client cannot contact a STUN server.

Note: The two latter options require the STUN client be enabled (EnableSTUN). Also, once the acKeepAlive trap is instigated it does not stop.

- The manager can view the NAT type in the MIB: audioCodes(5003).acProducts(9).acBoardMibs(10).acSystem(10).acSystemStatus(2).acSysNetwork(6).acSysNAT(2).acSysNATType(1).
- The manger also has access to the STUN client configuration: audioCodes(5003).acProducts(9).acBoardMibs(10).acSystem(10).acSystemConfiguration(1).acSysNetworkConfig(3).acSysNATTraversal(6).acSysSTUN(21).
- acNATTraversalAlarm: When the NAT is placed in front a device that is identified as a symmetric NAT, this alarm is raised. It is cleared when a non-symmetric NAT or no NAT replace the symmetric one.

16.11 SNMP Administrative State Control

16.11.1 Node Maintenance

Node maintenance for the Mediant 1000 is provided by an SNMP interface. The acBoardMIB provides two parameters for graceful and forced shutdowns of the Mediant 1000:

- acgwAdminState
- acgwAdminStateLockControl

The acgwAdminState is used either to request (set) a shutdown (0), undo shutdown (2), or to view (get) the gateway condition (0 = locked; 1 = shutting down; 2 = unlocked).

The acgwAdminStateLockControl is used to set a time limit (in seconds) for the shutdown where 0 means shutdown immediately (forced), -1 means no time limit (graceful), and x where x>0 indicates a time limit in seconds (timed limit is considered a graceful shutdown).

Note: The acgwAdminStateLockControl must be set first followed by the acgwAdminState.

16.11.2 Graceful Shutdown

acgwAdminState is a read-write MIB object. When a get request is sent for this object, the agent returns the board's current administrative state.

The possible values received on a **Get** request include the following:

- locked(0): the board is locked
- shuttingDown(1): the board is currently performing a graceful lock
- unlocked(2): the board is unlocked

On a **Set** request, the manager supplies one of the following desired administrative states:

- locked(0)
- unlocked(2)

When the board changes to either shuttingDown or locked state, an adminStateChange alarm is raised. When the board changes to an unlocked state, the adminStateChange alarm is cleared.

Before setting acgwAdminState to perform a lock, acgwAdminStateLockControl should be set first to control the type of lock that is performed. The possible values for the acgwAdminStateLockControl include the following:

- 1 = Perform a graceful lock. Calls are allowed to complete. No new calls are allowed to be originated on this device.
- 0 = Perform a forced lock. Calls are immediately terminated.
- Any number greater than 0 = Time in seconds before the graceful lock turns into a forced lock.

16.12 AudioCodes' Element Management System

Using AudioCodes' Element Management System (EMS) is recommended to Customers requiring large deployments (multiple media gateways in globally distributed enterprise offices, for example), that need to be managed by central personnel.

The EMS is not included in the device's supplied package. Contact AudioCodes for detailed information on AudioCodes' EMS and on AudioCodes' EVN - Enterprise VoIP Network – solution for large VoIP deployments.

17 Configuration Files

This section describes the configuration *dat* files that are loaded (in addition to the *ini* file) to the gateway. The configuration files include the following:

- Call Progress Tones file (refer to Section 17.1 on page 391)
- Prerecorded Tones file (refer to Section 17.2 on page 395)
- CAS protocol configuration files (refer to Section 17.3 on page 396)
- FXS/FXO Coefficient file (refer to Section 17.4 on page 396)
- User Information file (refer to Section 17.5 on page 397)

To load either of the configuration files to the Mediant 1000, use the Embedded Web Server (refer to Section 5.8.2 on page 196) or alternatively specify the name of the relevant configuration file in the gateway's *ini* file and load it (the *ini* file) to the gateway (refer to Section 5.8.2.2 on page 198).

17.1 Configuring the Call Progress Tones and Distinctive Ringing File

The Call Progress Tones and Distinctive Ringing, configuration file used by the Mediant 1000 is a binary file (with the extension *dat*) that is comprised of two sections. The first section contains the definitions of the Call Progress Tones (levels and frequencies) that are detected / generated by the Mediant 1000. The second section contains the characteristics of the distinctive ringing signals that are generated by the Mediant 1000.

Users can either use one of the supplied Mediant 1000 configuration (*dat*) files, or construct their own file. To construct their own configuration file, users are recommended, to modify the supplied *usa_tone.ini* file (in any standard text editor) to suit their specific requirements, and to convert it (the modified *ini* file) into binary format using the TrunkPack Downloadable Conversion Utility. For the description of the procedure on how to convert CPT *ini* file to a binary *dat* file, refer to Section G.1.1 on page 430.

Note: Only the *dat* file can be loaded to the Mediant 1000 gateway.

To load the Call Progress Tones *(dat*) file to the Mediant 1000, use the Embedded Web Server (refer to Section 5.8.2 on page 196) or the *ini* file (refer to Section 5.8.2.2 on page 198).

17.1.1 Format of the Call Progress Tones Section in the *ini* File

Users can create up to 32 different Call Progress Tones, each with frequency and format attributes.

The frequency attribute can be single or dual-frequency (in the range of 300 Hz to 1980 Hz), or an Amplitude Modulated (AM). In total, up to 64 different frequencies are supported. Only eight AM tones, in the range of 1 to 128 kHz, can be configured (the detection range is limited to 1 to 50 kHz). Note that when a tone is composed of a single frequency, the second frequency field must be set to zero.

The format attribute can be one of the following:

- **Continuous:** (e.g., dial tone) a steady non-interrupted sound. Only the 'First Signal On time' should be specified. All other on and off periods must be set to zero. In this case, the parameter specifies the detection period. For example, if it equals 300, the tone is detected after 3 seconds (300 x 10 msec). The minimum detection time is 100 msec.

- **Cadence:** A repeating sequence of on and off sounds. Up to four different sets of on / off periods can be specified.
- **Burst:** A single sound followed by silence. Only the ‘First Signal On time’ and ‘First Signal Off time’ should be specified. All other on and off periods must be set to zero. The burst tone is detected after the off time is completed.

Users can specify several tones of the same type. These additional tones are used only for tone detection. Generation of a specific tone conforms to the first definition of the specific tone. For example, users can define an additional dial tone by appending the second dial tone’s definition lines to the first tone definition in the *ini* file. The Mediant 1000 reports dial tone detection if either of the two tones is detected.

The Call Progress Tones section of the *ini* file comprises the following segments:

- **[NUMBER OF CALL PROGRESS TONES]** – Contains the following key: ‘Number of Call Progress Tones’ defining the number of Call Progress Tones that are defined in the file.
- **[CALL PROGRESS TONE #X]** – containing the Xth tone definition (starting from 1 and not exceeding the number of Call Progress Tones defined in the first section) using the following keys:
  - **Tone Type:** Call Progress Tone type

Figure 17-1: Call Progress Tone Types

```
1 - Dial Tone
2 - Ringback Tone
3 - Busy Tone
7 - Reorder Tone
8 - Confirmation Tone
9 - Call Waiting Tone
15 - Stutter Dial Tone
16 - Off Hook Warning Tone
17 - Call Waiting Ringback Tone
23 - Hold Tone
```

  - **Tone Modulation Type:** Either Amplitude Modulated (1) or regular (0).
  - **Tone Form:** The tone’s format, can be one of the following:
    - Continuous
    - Cadence
    - Burst
  - **Low Freq [Hz]:** Frequency in hertz of the lower tone component in case of dual frequency tone, or the frequency of the tone in case of single tone (not relevant to AM tones).
  - **High Freq [Hz:** Frequency in hertz of the higher tone component in case of dual frequency tone, or zero (0) in case of single tone (not relevant to AM tones).
  - **Low Freq Level [-dBm]:** Generation level 0 dBm to –31 dBm in [dBm] (not relevant to AM tones).
  - **High Freq Level:** Generation level. 0 to –31 dBm. The value should be set to ‘32’ in the case of a single tone (not relevant to AM tones).
  - **First Signal On Time [10 msec]:** ‘Signal On’ period (in 10 msec units) for the first cadence on-off cycle. For be continuous tones, this parameter defines the detection period. For burst tones, it defines the tone’s duration.
  - **First Signal Off Time [10 msec]:** ‘Signal Off’ period (in 10 msec units) for the first cadence on-off cycle (for cadence tones). For burst tones, this parameter defines the off time required after the burst tone ends and the tone detection is reported. For continuous tones, this parameter is ignored.
  - **Second Signal On Time [10 msec]:** ‘Signal On’ period (in 10 msec units) for the second cadence on-off cycle. Can be omitted if there isn’t a second cadence.

- **Second Signal Off Time [10 msec]:** 'Signal Off' period (in 10 msec units) for the second cadence on-off cycle. Can be omitted if there isn't a second cadence.
- ***Third Signal On Time [10 msec]***: 'Signal On' period (in 10 msec units) for the third cadence ON-OFF cycle. Can be omitted if there isn't a third cadence.
- ***Third Signal Off Time [10 msec]***: 'Signal Off' period (in 10 msec units) for the third cadence ON-OFF cycle. Can be omitted if there isn't a third cadence.
- ***Forth Signal On Time [10 msec]***: 'Signal On' period (in 10 msec units) for the fourth cadence ON-OFF cycle. Can be omitted if there isn't a fourth cadence.
- ***Forth Signal Off Time [10 msec]***: 'Signal Off' period (in 10 msec units) for the fourth cadence ON-OFF cycle. Can be omitted if there isn't a fourth cadence.
- **Carrier Freq [Hz]:** the frequency of the carrier signal for AM tones.
- **Modulation Freq [Hz]:** the frequency of the modulated signal for AM tones (valid range from 1 Hz to 128 Hz).
- **Signal Level [-dBm]:** the level of the tone for AM tones.
- **AM Factor [steps of 0.02]:** the amplitude modulation factor (valid range from 1 to 50. Recommended values from 10 to 25).

Note 1: When the same frequency is used for a continuous tone and a cadence tone, the 'Signal On Time' parameter of the continuous tone must have a value that is greater than the 'Signal On Time' parameter of the cadence tone. Otherwise the continuous tone is detected instead of the cadence tone.

Note 2: The tones frequency should differ by at least 40 Hz from one tone to other defined tones.

For example: to configure the dial tone to 440 Hz only, define the following text:

Figure 17-2: Defining a Dial Tone Example

```
#Dial tone
[CALL PROGRESS TONE #1]
Tone Type=1
Tone Form =1 (continuous)
Low Freq [Hz]=440
High Freq [Hz]=0
Low Freq Level [-dBm]=10 (-10 dBm)
High Freq Level [-dBm]=32 (use 32 only if a single tone is required)
First Signal On Time [10msec]=300; the dial tone is detected after 3 sec
First Signal Off Time [10msec]=0
Second Signal On Time [10msec]=0
Second Signal Off Time [10msec]=0
```

Figure 17-3: Example of Ringing Burst

```
#Three ringing bursts followed by repeated ringing of 1 sec on and 3 sec off.
[NUMBER OF DISTINCTIVE RINGING PATTERNS]
Number of Ringing Patterns=1
[Ringing Pattern #0]
Ring Type=0
Freq [Hz]=25
First Burst Ring On Time [10msec]=30
First Burst Ring Off Time [10msec]=30
Second Burst Ring On Time [10msec]=30
Second Burst Ring Off Time [10msec]=30
Third Burst Ring On Time [10msec]=30
Third Burst Ring Off Time [10msec]=30
Fourth Ring On Time [10msec]=100
Fourth Ring Off Time [10msec]=300
```

17.1.2 Format of the Distinctive Ringing Section in the *ini* File

Distinctive Ringing is only applicable to Mediant 1000/FXS modules. Using the distinctive ringing section of this configuration file, the user can create up to 16 distinctive ringing patterns.

Each ringing pattern configures the ringing tone frequency and up to 4 ringing cadences. The same ringing frequency is used for all the ringing pattern cadences. The ringing frequency can be configured in the range of 10 Hz to 200 Hz with a 5 Hz resolution. Each of the ringing pattern cadences is specified by the following parameters:

- **Burst Ring On Time:** Configures the cadence to be a burst cadence in the entire ringing pattern. The burst relates to On time and the Off time of the same cadence. It must appear between 'First/Second/Third/Fourth' string and the 'Ring On/Off Time' This cadence rings once during the ringing pattern. Otherwise, the cadence is interpreted as cyclic: it repeats for every ringing cycle.
- **Ring On Time:** specifies the duration of the ringing signal.
- **Ring Off Time:** specifies the silence period of the cadence.

Note: In SIP the distinctive ringing pattern is selected according to Alert-Info header that is included in INVITE message. For example: Alert-Info <Bellcore-dr2>, or Alert-Info<http://…/Bellcore-dr2>. 'dr2' defines ringing pattern # 2. If the Alert-Info header is missing, the default ringing tone (0) is played.

The distinctive ringing section of the *ini* file format contains the following strings:

- **[NUMBER OF DISTINCTIVE RINGING PATTERNS]:** Contains the following key:
  - 'Number of Distinctive Ringing Patterns' defining the number of Distinctive Ringing signals that are defined in the file.
- **[Ringing Pattern #X]:** Contains the Xth ringing pattern definition (starting from 0 and not exceeding the number of Distinctive Ringing patterns defined in the first section minus 1) using the following keys:
  - **Ring Type:** Must be equal to the Ringing Pattern number.
  - **Freq [Hz]:** Frequency in hertz of the ringing tone.
  - **First (Burst) Ring On Time [10 msec]:** 'Ring On' period (in 10 msec units) for the first cadence on-off cycle.
  - **First (Burst) Ring Off Time [10 msec]:** 'Ring Off' period (in 10 msec units) for the first cadence on-off cycle.
  - **Second (Burst) Ring On Time [10 msec]:** 'Ring On' period (in 10 msec units) for the second cadence on-off cycle.
  - **Second (Burst) Ring Off Time [10 msec]:** 'Ring Off' period (in 10 msec units) for the second cadence on-off cycle.
  - **Third (Burst) Ring On Time [10 msec]:** 'Ring On' period (in 10 msec units) for the third cadence on-off cycle.
  - **Third (Burst) Ring Off Time [10 msec]:** 'Ring Off' period (in 10 msec units) for the third cadence on-off cycle.
  - **Fourth (Burst) Ring On Time [10 msec]:** 'Ring Off' period (in 10 msec units) for the fourth cadence on-off cycle.
  - **Fourth (Burst) Ring Off Time [10 msec]:** 'Ring Off' period (in 10 msec units) for the fourth cadence on-off cycle.

17.1.2.1 Examples of Various Ringing Signals

Figure 17-4: Examples of Various Ringing Signals

```
[NUMBER OF DISTINCTIVE RINGING PATTERNS]
Number of Ringing Patterns=3

#Regular North American Ringing Pattern
[Ringing Pattern #0]
Ring Type=0
Freq [Hz]=20
First Ring On Time [10msec]=200
First Ring Off Time [10msec]=400

#GR-506-CORE Ringing Pattern 1
[Ringing Pattern #1]
Ring Type=1
Freq [Hz]=20
First Ring On Time [10msec]=200
First Ring Off Time [10msec]=400

#GR-506-CORE Ringing Pattern 2
[Ringing Pattern #2]
Ring Type=2
Freq [Hz]=20
First Ring On Time [10msec]=80
First Ring Off Time [10msec]=40
Second Ring On Time [10msec]=80
Second Ring Off Time [10msec]=400
```

17.2 Prerecorded Tones (PRT) File

The Call Progress Tones mechanism has several limitations, such as a limited number of predefined tones and a limited number of frequency integrations in one tone. To work around these limitations and provide tone generation capability that is more flexible, the PRT file can be used. If a specific prerecorded tone exists in the PRT file, it takes precedence over the same tone that exists in the CPT file and is played instead of it.

Note that the prerecorded tones are used only for generation of tones. Detection of tones is performed according to the CPT file.

17.2.1 PRT File Format

The PRT *dat* file contains a set of prerecorded tones to be played by the Mediant 1000 during operation. Up to 40 tones (totaling approximately one minute) can be stored in a single file in flash memory. The prerecorded tones (raw data PCM or L8 files) are prepared offline using standard recording utilities (such as CoolEdit^{TM}) and combined into a single file using the TrunkPack Downloadable Conversion utility (refer to Section G.1.4 on page 433).

The raw data files must be recorded with the following characteristics:

- Coders: G.711 A-law, G.711 µ-law or Linear PCM
- Rate: 8 kHz
- Resolution: 8-bit
- Channels: mono

The generated PRT file can then be loaded to the Mediant 1000 using the BootP/TFTP utility (refer to Section 5.8.2.2 on page 198) or via the Embedded Web Server (Section 5.8.2 on page 196).

The prerecorded tones are played repeatedly. This enables you to record only part of the tone and play it for the full duration. For example, if a tone has a cadence of 2 seconds on and 4 seconds off, the recorded file should contain only these 6 seconds. The PRT module repeatedly plays this cadence for the configured duration. Similarly, a continuous tone can be played by repeating only part of it.

17.3 CAS Protocol Configuration Files

The CAS Protocol Configuration Files contain the CAS Protocol definitions that are used for CAS-terminated trunks. You can either use the supplied files or construct your own files.

It is possible to load up to eight files and to use different files for different trunks.

Note: All CAS files loaded together must belong to the same Trunk Type (either E1 or T1).

17.4 The Coefficient Configuration File

The Coeff_FXS.dat file is used to provide best termination and transmission quality adaptation for different line types for FXS modules. This adaptation is performed by modifying the telephony interface characteristics (such as DC and AC impedance, feeding current and ringing voltage).

The *coeff.dat* configuration file is produced specifically for each market after comprehensive performance analysis and testing, and can be modified on request. The current file supports US line type of 600 ohm AC impedance and40 V RMS ringing voltage for REN = 2.

To load the coeff.dat file to the Mediant 1000, use the Embedded Web Server (refer to Section 5.8.2 on page 196) or alternatively specify the FXS coeff.dat file name in the gateway's *ini* file (refer to Section 5.8.2.2 on page 198).

The Coeff.dat file consists of a set of parameters for the signal processor of the loop interface devices. This parameter set provides control of the following AC and DC interface parameters:

- DC (battery) feed characteristics
- AC impedance matching
- Transmit gain
- Receive gain
- Hybrid balance
- Frequency response in transmit and receive direction
- Hook thresholds
- Ringing generation and detection parameters

This means, for example, that changing impedance matching or hybrid balance doesn't require hardware modifications, so that a single device is able to meet requirements for different markets. The digital design of the filters and gain stages also ensures high reliability, no drifts (over temperature or time) and simple variations between different line types.

In future software releases, it is to be expanded to consist of different sets of line parameters, which can be selected in the *ini* file, for each port.

17.5 User Information File

The User Information file maps PBX extensions (connected to the Mediant 1000 gateway) to global IP phone numbers (alphanumerical). In this context, a global IP number serves as a routing identifier for calls in the 'IP World'. The PBX extension uses this mapping to emulate the behavior of an IP phone.

Note: The mapping mechanism is disabled by default and must be activated using the parameter 'EnableUserInfoUsage' (described in Section 5.5.2.1 on page 83).

Each line in the file represents a mapping rule of a single PBX extension (up to 100 rules can be configured). Each line includes five items separated with commas. The items are described in Table 17-1 below. An example of a User Information file is shown in Figure 17-5 below.

Each PBX extension registers separately (a REGISTER message is sent for each entry, only if AuthenticationMode is set to 'Per Endpoint') using the IP number in the From / To headers. The REGISTER messages are sent gradually (i.e., initially, the gateway sends requests according to the maximum number of allowed SIP dialogs (configured by the parameter NumberOfActiveDialogs), after each received response, the subsequent request is sent). Therefore, no more than 'NumberOfActiveDialogs' dialogs are active simultaneously. The username and password are used for SIP Authentication when required.

The calling number of outgoing Tel to IP calls is first translated to an IP number and then (if defined) the manipulation rules are performed. The Display Name is used in the From header in addition to the IP number.

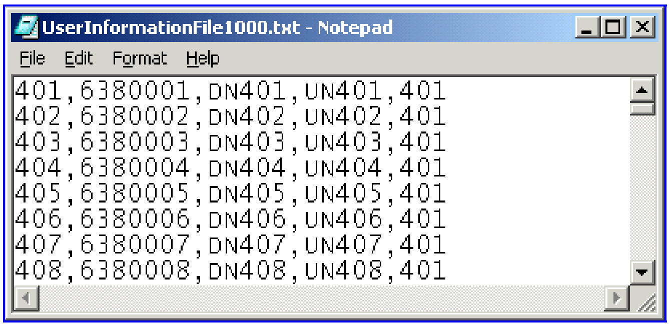
The called number of incoming IP to Tel calls is translated to a PBX extension only after manipulation rules (if defined) are performed.

The User Information file is a text file (the file size mustn't exceed 10,800 bytes for analog modules and 108,000 bytes for digital modules) that can be loaded via the *ini* file (UserInfoFileName, described in Section 6.5.14 on page 258), the Embedded Web Server (refer to Section 5.8.2 on page 196) or by using the automatic update mechanism (UserInfoFileURL, refer to Section 12.3 on page 320).

Table 17-1: User Information Items

Item	Description	Maximum Size
PBX extension #	The relevant PBX extension number	10
Global IP #	The relevant IP phone number	20
Display name	A string that represents the PBX extensions for the Caller ID.	30
Username	A string that represents the username for SIP registration.	20
Password	A string that represents the password for SIP registration.	20

Figure 17-5: Example of a User Information File



```
401,6380001,DN401,UN401,401
402,6380002,DN402,UN402,401
403,6380003,DN403,UN403,401
404,6380004,DN404,UN404,401
405,6380005,DN405,UN405,401
406,6380006,DN406,UN406,401
407,6380007,DN407,UN407,401
408,6380008,DN408,UN408,401
```

Reader's Notes

A Selected Technical Specifications

Table A-1: Mediant 1000 Functional Specifications (continues on pages 399 to 402)

Modularity and Capacity[3]	
	Six slots for analog modules.Up to 24 FXS/FXO analog ports. Up to four digital trunks (fully flexible, from a single and up to four trunks per module)
Interface I/O Modules	
FXS Telephony Interface	Up to 6 modules with 4 FXS RJ-11 ports on each module (for a total of up to 24 analog FXS RJ-11 ports).
FXO Telephony Interface	Up to 6 modules with 4 FXO RJ-11 ports on each module (for a total of up to 24 analog FXO RJ-11 ports).
Digital Modules	1, 2, 3 or 4 E1/T1/J1 spans (Balanced 120/100 ohm) using RJ-48 connectors per module. Up to 4 digital modules (maximum 4 spans per gateway). Optional 1+1 or 2+2 fallback.
FXS Functionality	
FXS Capabilities	Short or long haul, up to 3,000 m (10,000 ft.), using 24 AWG line cord.
	Number of ports per FXS module: 2 or 4
	Caller ID generation: Bellcore GR-30-CORE Type 1 using Bell 202 FSK modulation, ETSI Type 1, NTT, Denmark, India, Brazil, British and DTMF ETSI CID (ETS 300-659-1).
	Polarity Reversal and Wink signals generation
	Message waiting indication (lamp)
	Programmable Line Characteristics: Battery feed, line current, hook thresholds, AC impedance matching, hybrid balance, Tx & Rx frequency response, Tx & Rx Gains **Note:** For a specific coefficient file contact AudioCodes.
	Programmable ringing signal. Up to three cadences and frequency 10 to 200 Hz.
	Drive 4 phones per port simultaneously in offhook and Ring states. REN = 5.
	Over-temperature protection for abnormal situations such as shorted lines.
	Lifeline, on every FXS factory-preconfigured module.
FXO Functionality	
FXO Capabilities	Short or long haul, up to 7,000 m (24,000 ft.), using 24 AWG line cord.
	Number of ports per FXO module: 4
	Far end disconnect detection.
	Includes lightning and high voltage protection for outdoor operation
	Programmable Line Characteristics: AC impedance matching, hybrid balance, Tx & Rx frequency response, Tx & Rx Gains, ring detection threshold, DC characteristics **Note:** For a specific coefficient file contact AudioCodes.
	Caller ID Detection: Caller ID detection: Bellcore GR-30-CORE Type 1 using Bell 202 FSK modulation, ETSI Type 1, NTT, Denmark, India, Brazil, British and DTMF ETSI CID (ETS 300-659-1).
	Polarity Reversal and Wink signal detection

[3] Mediant 1000 channel capacity depends on configuration settings.

Table A-1: Mediant 1000 Functional Specifications (continues on pages 399 to 402)

Voice & Tone Characteristics	
Voice Compression	G.711 PCM at 64 kbps µ-law/A-law (10, 20, 30, 40, 50, 60, 80, 100, 120 msec) G.723.1 MP-MLQ at 5.3 or 6.3 kbps (30, 60, 90 msec) G.726 at 32 kbps ADPCM (10, 20, 30, 40, 50, 60, 80, 100, 120 msec) G.729 CS-ACELP 8 Kbps Annex A / B (10, 20, 30, 40, 50, 60 msec) NetCoder at 6.4, 7.2, 8.0 and 8.8 kbps (20, 40, 60, 80, 100, 120 msec). Transparent (20, 40, 60, 80, 100, 120 msec) Microsoft GSM (40 msec)
Silence Suppression	G.723.1 Annex A G.729 Annex B PCM and ADPCM - Standard Silence Descriptor (SID) with Proprietary Voice Activity Detection (VAD) and Comfort Noise Generation (CNG).
Packet Loss Concealment	G.711 appendix 1 G.723.1 G.729 a/b
Echo Canceler	G.165 and G.168 2000, 64 msec
Gain Control	Programmable
DTMF Transport (in-band)	Mute, transfer in RTP payload or relay in compliance with RFC 2833
DTMF Detection and Generation	Dynamic range 0 to -25 dBm, compliant with TIA 464B and Bellcore TR-NWT-000506.
Call Progress Tone Detection and Generation	32 tones: single tone, dual tones or AM tones, programmable frequency & amplitude; 64 frequencies in the range 300 to 1980 Hz, 1 to 4 cadences per tone, up to 4 sets of ON/OFF periods.
Output Gain Control	-32 dB to +31 dB in steps of 1 dB
Input Gain Control	-32 dB to +31 dB in steps of 1 dB
Conferencing	
Module	Optional Conference module housed in Slot 6 on the chassis front panel.
Conference Channels (Max.)	60
Simultaneous 3-Way Conferences (Max.)	20
Full-duplex parties per conference bridge ((Max.)	60
Fax/Modem Relay	
Fax Relay	Group 3 fax relay up to 14.4 kbps with auto fallback T.30 (PSTN) and T.38 (IP) compliant, real time fax relay Tolerant network delay (up to 9 seconds round trip) CNG tone detection & Relay per T.38 Answer tone (CED or AnsAm) detection & Relay per T.38
Fax Transparency	Automatic fax bypass (pass-through) to G.711, ADPCM or NSE bypass mode
Modem Transparency	Auto switch to PCM or ADPCM on V.34 or V.90 modem detection Automatic switching (pass-through) to PCM, ADPCM or NSE bypass mode for modem signals (V.34 or V.90 modem detection)

Table A-1: Mediant 1000 Functional Specifications (continues on pages 399 to 402)

Protocols	
VoIP Signaling Protocol	SIP RFC 3261
Communication Protocols	RTP/RTCP packetization. IP stack (UDP, TCP, and RTP). Remote Software load (TFTP, HTTP and HTTPS).
Telephony Protocols	PRI (ETSI Euro ISDN, ANSI NI2, 4/5ESS, DMS 100, QSIG, Japan INS1500, Australian Telecom, New Zealand Telecom, Hong Kong Variant, Korean MIC) E1/T1 CAS protocols: MFC R2, E&M wink start, Immediate start, delay start, loop start, ground start, Feature Group B, D for E1/T1
In-Band Signaling	DTMF (TIA 464A) MF-R1, MFC R2 User-defined Call Progress Tones
Line Signaling Protocols	Loop start and ground start
CPU	
Network Interface	Two Ethernet RJ-45 connectors, 10/100 Base-TX
RS-232 Interface	RS-232 terminal interface. Non-standard RS-232 connector on the device’s CPU.
Reset	Resets the device.
Dry Contact	NB (Night Bell) and paging
Audio I/O	MOH (Music on Hold) and paging
Physical	
Dimensions	19” (width) by 1U (height) by 13.8” (depth)
Weight	Approx. 5 kg (depends on number of installed modules)
Supply Voltage and Power Consumption	Universal 100 - 240 VAC; 50 - 60 Hz; 1 A max.
Environmental	Operational: 0° to 45° C 32° to 113° F Storage: -10° to 70° C 14° to 158° F Humidity: 10 to 90% non-condensing
Installation	Standard 19-inch rack mount or shelf
OSN Server	
Single Chassis Integration	Embedded Pentium^{TM} Celeron^{TM} based platform for third-party hosted applications
CPU	Intel TM Pentium-M Celeron ULV 600 MHz processor
	855 GME + 6300 ESB Intel chipset
	128 KB BIOS flash
	Operational 0° to 40° C
	160 mm (6.4”) depth x 75 mm (2.9”) width x 30 mm (1.17”) height
	Up to 1 Giga Byte 200/266 MHz SODIMM memory.
HHD	40 Giga Byte Hard Disk Drive 5200 RPM (second HDD optional)
Interfaces	10/100 Base-TX, USB, RS-232, NB relay, MOH
Diagnostics	
Front panel Status LEDs	E1/T1 status LAN status Gateway status (Fail, ACT, Power, and Swap Ready).
Syslog events	Supported by Syslog Server, per RFC 3164 IETF standard.
SNMP MIBs and Traps	SNMP v2c, SNMP v3

Table A-1: Mediant 1000 Functional Specifications (continues on pages 399 to 402)

Management	
Configuration	Gateway configuration using Web browser, CLI or *ini* files
Management and Maintenance	SNMP v2c, SNMP v3
	Syslog, per RFC 3164
	Local RS-232 terminal
	Web Management via HTTP or HTTPS
	Telnet

B Supplied SIP Software Kit

Table B-1 describes the standard supplied software kit for Mediant 1000. The supplied documentation includes this User's Manual, the Mediant 1000 MEGACO-H.323-SIP Fast Track Guide and the Mediant 1000 SIP Release Notes.

Table B-1: Mediant 1000 SIP Supplied Software Kit

File Name	Description
Ram.cmp files	
M1000_SIP_xxx.cmp	Image file containing the software for both FXS and FXO modules.
M1000_Digital_SIP_xxx.cmp	Image file containing the software for the Mediant 1000 gateway.
***ini* files and utilities**	
SIPgw_M1K.ini	Sample *Ini* file for the Mediant 1000 media gateway.
M1000_Digital_SIP_T1.ini	Sample *ini* file for Mediant 1000 E1 gateways.
M1000_Digital_SIP_E1.ini	Sample *ini* file for Mediant 1000 T1 gateways.
Usa_tones_xx.dat	Default loadable Call Progress Tones *dat* file.
Usa_tones_xx.ini	Call progress Tones *ini* file (used to create *dat* file).
Coeff_FXS.dat	Telephony interface configuration file for FXS modules.
Coeff_FXO.dat	Telephony interface configuration file for FXO modules.
DConvert240.exe	TrunkPack Downloadable Conversion Utility
ACSyslog08.exe	Syslog server.
bootp.exe	BootP/TFTP configuration utility
CPTWizard.exe	Call Progress Tones Wizard
CAS Protocol Files	Used for various signaling types, such as *E_M_WinkTable.dat.*
MIBs Files	MIB library for SNMP browser
CAS Capture Tool	Utility that is used to convert CAS traces to textual form.
ISDN Capture Tool	Utility that is used to convert ISDN traces to textual form.

Reader's Notes

C SIP Compliance Tables

The gateway complies with RFC 3261, as shown in the following sections.

C.1 SIP Functions

Table C-1: SIP Functions

Function	Supported
User Agent Client (UAC)	Yes
User Agent Server (UAS)	Yes
Proxy Server	Third-party only (verified with Ubiquity, Delta3, Microsoft, 3Com, BroadSoft, Snom and Cisco Proxies amongst others)
Redirect Server	Third-party
Registrar Server	Third-party
Event Publication Agent (EPA)	Yes
Event State Compositor (ESC)	Third-party

C.2 SIP Methods

Table C-2: SIP Methods

Method	Supported	Comments
INVITE	Yes	
ACK	Yes	
BYE	Yes	
CANCEL	Yes	
REGISTER	Yes	Send only
REFER	Yes	
NOTIFY	Yes	
INFO	Yes	
OPTIONS	Yes	
PRACK	Yes	
UPDATE	Yes	
PUBLISH	Yes	
SUBSCRIBE	Yes	

C.3 SIP Headers

The following SIP Headers are supported by the gateway:

Table C-3: SIP Headers (continues on pages 406 to 407)

Header Field	Supported
Accept	Yes
Accept–Encoding	Yes
Alert-Info	Yes
Allow	Yes
Also	Yes
Asserted-Identity	Yes
Authorization	Yes
Call-ID	Yes
Call-Info	Yes
Contact	Yes
Content-Disposition	Yes
Content-Encoding	Yes
Content-Length	Yes
Content-Type	Yes
Cseq	Yes
Date	Yes
Diversion	Yes
Encryption	No
Expires	Yes
Fax	Yes
From	Yes
History-Info	Yes
Join	Yes
Max-Forwards	Yes
Messages-Waiting	Yes
MIN-SE	Yes
Organization	No
P-Asserted-Identity	Yes
P-Preferred-Identity	Yes
Priority	No
Proxy- Authenticate	Yes
Proxy- Authorization	Yes
Proxy- Require	Yes
Prack	Yes
Reason	Yes
Record- Route	Yes
Refer-To	Yes
Referred-By	Yes
Replaces	Yes
Require	Yes
Remote-Party-ID	Yes

Table C-3: SIP Headers (continues on pages 406 to 407)

Header Field	Supported
Response- Key	Yes
Retry- After	Yes
Route	Yes
Rseq	Yes
Session-Expires	Yes
Server	Yes
SIP-If-Match	Yes
Subject	Yes
Supported	Yes
Timestamp	Yes
To	Yes
Unsupported	Yes
User- Agent	Yes
Via	Yes
Voicemail	Yes
Warning	Yes
WWW- Authenticate	Yes

C.4 SDP Headers

The following SDP Headers are supported by the gateway:

Table C-4: SDP Headers

SDP Header Element	Supported
v - Protocol version	Yes
o - Owner/ creator and session identifier	Yes
a - Attribute information	Yes
c - Connection information	Yes
d - Digit	Yes
m - Media name and transport address	Yes
s - Session information	Yes
t - Time alive header	Yes
b - Bandwidth header	Yes
u - Uri Description Header	Yes
e - Email Address header	Yes
i - Session Info Header	Yes
p - Phone number header	Yes
y - Year	Yes

C.5 SIP Responses

The following SIP responses are supported by the gateway:

- 1xx Response - Information Responses
- 2xx Response - Successful Responses
- 3xx Response - Redirection Responses
- 4xx Response - Client Failure Responses
- 5xx Response - Server Failure Responses
- 6xx Response - Global Responses

C.5.1 1xx Response – Information Responses

Table C-5: 1xx SIP Responses

1xx Response		Supported	Comments
100	Trying	Yes	The SIP gateway generates this response upon receiving of Proceeding message from ISDN or immediately after placing a call for CAS signaling.
180	Ringing	Yes	The SIP gateway generates this response for an incoming INVITE message. On receiving this response, the gateway waits for a 200 OK response.
181	Call is being forwarded	Yes	The SIP gateway does not generate these responses. However, the gateway does receive them. The gateway processes these responses the same way that it processes the 100 Trying response.
182	Queued	Yes	The SIP gateway generates this response in Call Waiting service. When SIP gateway receives a 182 response, it plays a special waiting Ringback tone to TEL side.
183	Session Progress	Yes	The SIP gateway generates this response if Early Media feature is enabled and if the gateway plays a Ringback tone to IP

C.5.2 2xx Response – Successful Responses

Table C-6: 2xx SIP Responses

2xx Response		Supported	Comments
200	OK	Yes	
202	Accepted	Yes	

C.5.3 3xx Response – Redirection Responses

Table C-7: 3xx SIP Responses

3xx Response		Supported	Comments
300	Multiple Choice	Yes	The gateway responds with an ACK and resends the request to first in the contact list, new address.
301	Moved Permanently	Yes	The gateway responds with an ACK and resends the request to new address.
302	Moved Temporarily	Yes	The SIP gateway generates this response when call forward is used, to redirect the call to another destination. If such response is received, the calling gateway initiates an INVITE message to the new destination.
305	Use Proxy	Yes	The gateway responds with an ACK and resends the request to new address.
380	Alternate Service	Yes	"

C.5.4 4xx Response – Client Failure Responses

Table C-8: 4xx SIP Responses (continues on pages 409 to 410)

4xx Response		Supported	Comments
400	Bad Request	Yes	The gateway does not generate this response. On reception of this message, before a 200 OK has been received, the gateway responds with an ACK and disconnects the call.
401	Unauthorized	Yes	Authentication support for Basic and Digest. On receiving this message the GW issues a new request according to the scheme received on this response
402	Payment Required	Yes	The gateway does not generate this response. On reception of this message, before a 200 OK has been received, the gateway responds with an ACK and disconnects the call.
403	Forbidden	Yes	The gateway does not generate this response. On reception of this message, before a 200 OK has been received, the gateway responds with an ACK and disconnects the call.
404	Not Found	Yes	The SIP gateway generates this response if it is unable to locate the callee. On receiving this response, the gateway notifies the User with a Reorder Tone.
405	Method Not Allowed	Yes	The gateway does not generate this response. On reception of this message, before a 200OK has been received, the gateway responds with an ACK and disconnects the call.
406	Not Acceptable	Yes	The gateway does not generate this response. On reception of this message, before a 200OK has been received, the gateway responds with an ACK and disconnects the call.
407	Proxy Authentication Required	Yes	Authentication support for Basic and Digest. On receiving this message the GW issues a new request according to the scheme received on this response.
408	Request Timeout	Yes	The gateway generates this response if the no-answer timer expires. On reception of this message, before a 200OK has been received, the gateway responds with an ACK and disconnects the call.

Table C-8: 4xx SIP Responses (continues on pages 409 to 410)

4xx Response		Supported	Comments
409	Conflict	Yes	The gateway does not generate this response. On reception of this message, before a 200OK has been received, the gateway responds with an ACK and disconnects the call.
410	Gone	Yes	The gateway does not generate this response. On reception of this message, before a 200OK has been received, the gateway responds with an ACK and disconnects the call.
411	Length Required	Yes	The gateway does not generate this response. On reception of this message, before a 200OK has been received, the gateway responds with an ACK and disconnects the call.
413	Request Entity Too Large	Yes	The gateway does not generate this response. On reception of this message, before a 200OK has been received, the gateway responds with an ACK and disconnects the call.
414	Request-URL Too Long	Yes	The gateway does not generate this response. On reception of this message, before a 200OK has been received, the gateway responds with an ACK and disconnects the call.
415	Unsupported Media	Yes	If the gateway receives a 415 Unsupported Media response, it notifies the User with a Reorder Tone. The gateway generates this response in case of SDP mismatch.
420	Bad Extension	Yes	The gateway does not generate this response. On reception of this message, before a 200OK has been received, the gateway responds with an ACK and disconnects the call.
480	Temporarily Unavailable	Yes	If the gateway receives a 480 Temporarily Unavailable response, it notifies the User with a Reorder Tone. This response is issued if there is no response from remote.
481	Call Leg/Transaction Does Not Exist	Yes	The gateway does not generate this response. On reception of this message, before a 200OK has been received, the gateway responds with an ACK and disconnects the call.
482	Loop Detected	Yes	The gateway does not generate this response. On reception of this message, before a 200OK has been received, the gateway responds with an ACK and disconnects the call.
483	Too Many Hops	Yes	The gateway does not generate this response. On reception of this message, before a 200OK has been received, the gateway responds with an ACK and disconnects the call.
484	Address Incomplete	Yes	The gateway does not generate this response. On reception of this message, before a 200OK has been received, the gateway responds with an ACK and disconnects the call.
485	Ambiguous	Yes	The gateway does not generate this response. On reception of this message, before a 200OK has been received, the gateway responds with an ACK and disconnects the call.
486	Busy Here	Yes	The SIP gateway generates this response if the called party is off hook and the call cannot be presented as a call waiting call. On receiving this response, the gateway notifies the User and generates a busy tone.
487	Request Canceled	Yes	This response indicates that the initial request is terminated with a BYE or CANCEL request.
488	Not Acceptable	Yes	The gateway does not generate this response. On reception of this message, before a 200OK has been received, the gateway responds with an ACK and disconnects the call.

C.5.5 5xx Response – Server Failure Responses

Table C-9: 5xx SIP Responses

5xx Response		Comments
500	Internal Server Error	On reception of any of these Responses, the GW releases the call, sending appropriate release cause to PSTN side. The GW generates 5xx response according to PSTN release cause coming from PSTN.
501	Not Implemented	
502	Bad gateway	
503	Service Unavailable	
504	Gateway Timeout	
505	Version Not Supported	

C.5.6 6xx Response – Global Responses

Table C-10: 6xx SIP Responses

6xx Response		Comments
600	Busy Everywhere	On reception of any of these Responses, the GW releases the call, sending appropriate release cause to PSTN side.
603	Decline	
604	Does Not Exist Anywhere	
606	Not Acceptable	

Reader's Notes

D BootP/TFTP Configuration Utility

The BootP/TFTP utility enables you to easily configure and provision our boards and media gateways. Similar to third-party BootP/TFTP utilities (which are also supported) but with added functionality; our BootP/TFTP utility can be installed on Windows™ 98 or Windows™ NT/2000/XP. The BootP/TFTP utility enables remote reset of the device to trigger the initialization procedure (BootP and TFTP). It contains BootP and TFTP utilities with specific adaptations to our requirements.

D.1 When to Use the BootP/TFTP

The BootP/TFTP utility can be used with the device as an alternative means of initializing the gateways. Initialization provides a gateway with an IP address, subnet mask, and the default gateway IP address. The tool also loads default software, *ini* and other configuration files. BootP Tool can also be used to restore a gateway to its initial configuration, such as in the following instances:

- The IP address of the gateway is not known.
- The Web browser has been inadvertently turned off.
- The Web browser password has been forgotten.
- The gateway has encountered a fault that cannot be recovered using the Web browser.

Tip: The BootP is normally used to configure the device’s initial parameters. Once this information has been provided, the BootP is no longer needed. All parameters are stored in non-volatile memory and used when the BootP is not accessible.

D.2 An Overview of BootP

BootP is a protocol defined in RFC 951 and RFC 1542 that enables an internet device to discover its own IP address and the IP address of a BootP on the network, and to obtain the files from that utility that need to be loaded into the device to function.

A device that uses BootP when it powers up broadcasts a BootRequest message on the network. A BootP on the network receives this message and generates a BootReply. The BootReply indicates the IP address that should be used by the device and specifies an IP address from which the unit may load configuration files using Trivial File Transfer Protocol (TFTP) described in RFC 906 and RFC 1350.

D.3 Key Features

- Internal BootP supporting hundreds of entities.
- Internal TFTP.
- Contains all required data for our products in predefined format.
- Provides a TFTP address, enabling network separation of TFTP and BootP utilities.
- Tools to backup and restore the local database.
- Templates.
- User-defined names for each entity.
- Option for changing MAC address.
- Protection against entering faulty information.
- Remote reset.

- Unicast BootP response.
- User-initiated BootP respond, for remote provisioning over WAN.
- Filtered display of BootP requests.
- Location of other BootP utilities that contain the same MAC entity.
- Common log window for both BootP and TFTP sessions.
- Works with Windows™ 98, Windows™ NT, Windows™ 2000 and Windows™ XP.

D.4 Specifications

- BootP standards: RFC 951 and RFC 1542
- TFTP standards: RFC 1350 and RFC 906
- Operating System: Windows™ 98, Windows™ NT, Windows™ 2000 and Windows™ XP
- Max number of MAC entries: 200

D.5 Installation

➢ To install the BootP/TFTP on your computer, take these 2 steps:

1. Locate the BootP folder on the VoIP gateway supplied CD ROM and open the file Setup.exe.
2. Follow the prompts from the installation wizard to complete the installation.

➢ To open the BootP/TFTP, take these 2 steps:

1. From the **Start** menu on your computer, navigate to **Programs** and then click on **BootP**.
2. The first time that you run the BootP/TFTP, the program prompts you to set the user preferences. Refer to the Section D.10 on page 417 for information on setting the preferences.

D.6 Loading the *cmp* File, Booting the Device

Once the application is running, and the preferences were set (refer to Section D.10), for each unit that is to be supported, enter parameters into the tool to set up the network configuration information and initialization file names. Each unit is identified by a MAC address. For information on how to configure (add, delete and edit) units, refer to Section D.11 on page 419.

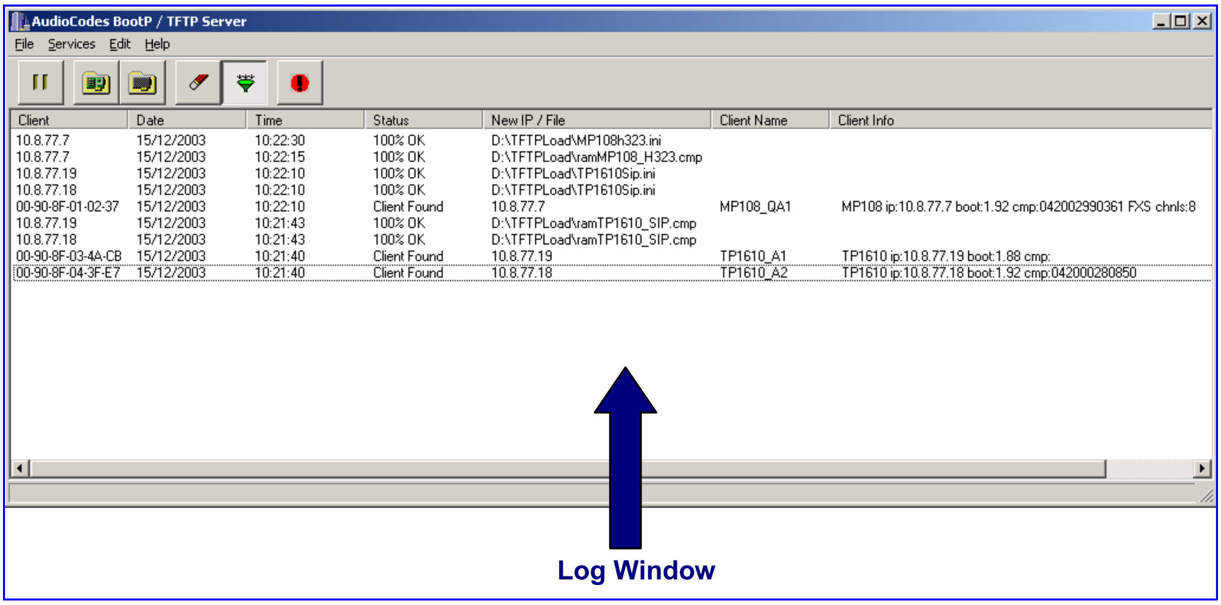
➢ To load the software and configuration files, take these 4 steps:

1. Create a folder on your computer that contains all software and configuration files that are needed as part of the TFTP process.
2. Set the BootP and TFTP preferences (refer to Section D.10).
3. Add client configuration for the VoIP gateway that you want to initialize by the BootP, refer to Section D.11.1.
4. Reset the VoIP gateway, either physically or remotely, causing the device to use BootP to access the network and configuration information.

D.7 BootP/TFTP Application User Interface

Figure D-1 shows the main application screen for the BootP/TFTP utility.

Figure D-1: Main Screen



AudioCodes BootP / TFTP Server

Client	Date	Time	Status	New IP / File	Client Name	Client Info
10.8.77.7	15/12/2003	10:22:30	100% OK	D:\TFTPLoad\MP108h323.ini		
10.8.77.7	15/12/2003	10:22:15	100% OK	D:\TFTPLoad\ramMP108_H323.cmp		
10.8.77.19	15/12/2003	10:22:10	100% OK	D:\TFTPLoad\TP1610Sip.ini		
10.8.77.18	15/12/2003	10:22:10	100% OK	D:\TFTPLoad\TP1610Sip.ini		
00-90-8F-01-02-37	15/12/2003	10:22:10	Client Found	10.8.77.7	MP108_QA1	MP108 ip:10.8.77.7 boot:1.92 cmp:042002990361 FXS chnls:8
10.8.77.19	15/12/2003	10:21:43	100% OK	D:\TFTPLoad\ramTP1610_SIP.cmp		
10.8.77.18	15/12/2003	10:21:43	100% OK	D:\TFTPLoad\ramTP1610_SIP.cmp		
00-90-8F-03-4A-CB	15/12/2003	10:21:40	Client Found	10.8.77.19	TP1610_A1	TP1610 ip:10.8.77.19 boot:1.88 cmp:
00-90-8F-04-3F-E7	15/12/2003	10:21:40	Client Found	10.8.77.18	TP1610_A2	TP1610 ip:10.8.77.18 boot:1.92 cmp:042000280850

Log Window

D.8 Function Buttons on the Main Screen

Pause: Click this button to pause the BootP Tool so that no replies are sent to BootP requests. Click the button again to restart the BootP Tool so that it responds to all BootP requests. The **Pause** button provides a depressed graphic when the feature is active.

Edit Clients: Click this button to open a new window that enables you to enter configuration information for each supported VoIP gateway. Details on the Clients window are provided in Section D.11 on page 419.

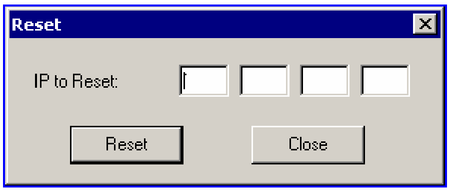
Edit Templates: Click this button to open a new window that enables you to create or edit standard templates. These templates can be used when configuring new clients that share most of the same settings. Details on the **Templates** window are provided in Section D.12 on page 424.

Clear Log: Click this button to clear all entries from the Log Window portion of the main application screen. Details on the log window are provided in Section D.9 on page 416.

Filter Clients: Click this button to prevent the BootP Tool from logging BootP requests received from disabled clients or from clients which do not have entries in the Clients table.

Reset: Click this button to open a new window where you enter an IP address requests for a gateway that you want to reset.

Figure D-2: Reset Screen



When a gateway resets, it first sends a BootRequest. Therefore, Reset can be used to force a BootP session with a gateway without needing to power cycle the gateway. As with any BootP session, the computer running the BootP Tool must be located on the same subnet as the controlled VoIP gateway.

D.9 Log Window

The log window (refer to Figure D-1 on the previous page) records all BootP request and BootP reply transactions, as well as TFTP transactions. For each transaction, the log window displays the following information:

- **Client:** shows the Client address of the VoIP gateway, which is the MAC address of the client for BootP transactions or the IP address of the client for TFTP transactions.
- **Date:** shows the date of the transaction, based on the internal calendar of the computer.
- **Time:** shows the time of day of the transaction, based on the internal clock of the computer.
- **Status:** indicates the status of the transaction.
  - *Client Not Found:* A BootRequest was received but there is no matching client entry in the BootP Tool.
  - *Client Found:* A BootRequest was received and there is a matching client entry in the BootP Tool. A BootReply is sent.
  - *Client's MAC Changed:* There is a client entered for this IP address but with a different MAC address.
  - *Client Disabled:* A BootRequest was received and there is a matching client entry in the BootP tool but this entry is disabled.
  - *Listed At:* Another BootP utility is listed as supporting a particular client when the Test Selected Client button is clicked (for details on Testing a client, refer to Section D.11.4 on page 420).
  - *Download Status:* Progress of a TFTP load to a client, shown in %.
- **New IP / File:** shows the IP address applied to the client as a result of the BootP transaction, as well as the file name and path of a file transfer for a TFTP transaction.
- **Client Name:** shows the client name, as configured for that client in the Client Configuration screen.

Use right-click on a line in the Log Window to open a pop-up window with the following options:

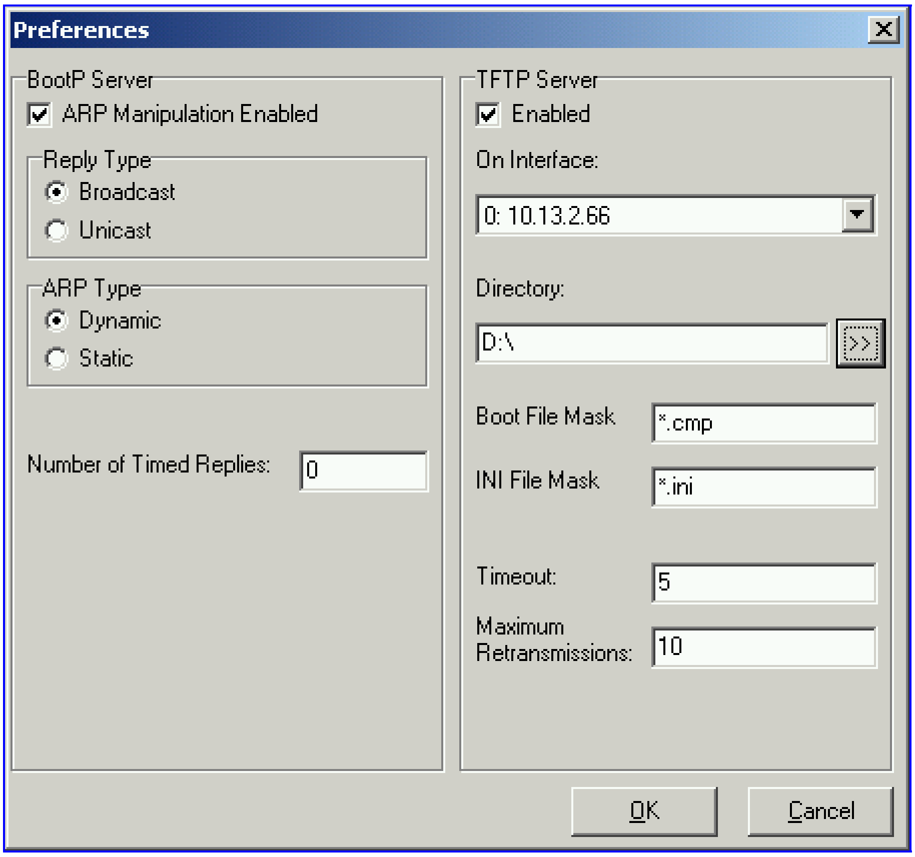
- **Reset:** Selecting this option results in a reset command being sent to the client VoIP gateway. The program searches its database for the MAC address indicated in the line. If the client is found in that database, the program adds the client MAC address to the Address Resolution Protocol (ARP) table for the computer. The program then sends a reset command to the client. This enables a reset to be sent without knowing the current IP address of the client, as long as the computer sending the reset is on the same subnet.
  Note: In order to use reset as described above, the user must have administrator privileges on the computer. Attempting to perform this type of reset without administrator privileges on the computer results in an error message. **ARP Manipulation Enable** must also be turned on in the **Preferences** window.

- **View Client:** Selecting this option, or double clicking on the line in the log window, opens the **Client Configuration** window. If the MAC address indicated on the line exists in the client database, it is highlighted. If the address is not in the client database, a new client is added with the MAC address filled out. You can enter data in the remaining fields to create a new client entry for that client.

D.10 Setting the Preferences

The Preferences window (refer to Figure D-3) is used to configure the BootP Tool parameters.

Figure D-3: Preferences Screen



Preferences

BootP Server

☑ ARP Manipulation Enabled

Reply Type

● Broadcast

○ Unicast

ARP Type

● Dynamic

○ Static

Number of Timed Replies: 0

TFTP Server

☑ Enabled

On Interface: 0: 10.13.2.66

Directory: D:\

Boot File Mask *.cmp

INI File Mask *.ini

Timeout: 5

Maximum Retransmissions: 10

D.10.1 BootP Preferences

ARP is a common acronym for Address Resolution Protocol, and is the method used by all Internet devices to determine the link layer address, such as the Ethernet MAC address, in order to route Datagrams to devices that are on the same subnet.

When ARP Manipulation is enabled on this screen, the BootP Tool creates an ARP cache entry on your computer when it receives a BootP BootRequest from the VoIP gateway. Your computer uses this information to send messages to the VoIP gateway without using ARP again. This is particularly useful when the gateway does not yet have an IP address and, therefore, cannot respond to an ARP.

Because this feature creates an entry in the computer ARP cache, Administrator Privileges are required. If the computer is not set to allow administrator privileges, ARP Manipulation cannot be enabled.

- **ARP Manipulation Enabled:** Enable ARP Manipulation to remotely reset a gateway that does not yet have a valid IP address.

  If ARP Manipulation is enabled, the following two commands are available.

  - **Reply Type:** Reply to a BootRequest can be either **Broadcast** or **Unicast**. The default for the BootP Tool is **Broadcast**. In order for the reply to be set to **Unicast**, ARP Manipulation must first be enabled. This then enables the BootP Tool to find the MAC address for the client in the ARP cache so that it can send a message directly to the requesting device. Normally, this setting can be left at **Broadcast**.
  - **ARP Type:** The type of entry made into the ARP cache on the computer, once **ARP Manipulation** is enabled, can be either **Dynamic** or **Static**. Dynamic entries expire after a period of time, keeping the cache clean so that stale entries do not consume computer resources. The Dynamic setting is the default setting and the setting most often used. Static entries do not expire.

- **Number of Timed Replies:** This feature is useful for communicating to VoIP gateways that are located behind a firewall that would block their BootRequest messages from getting through to the computer that is running the BootP Tool. You can set this value to any whole digit. Once set, the BootP Tool can send that number of BootReply messages to the destination immediately after you send a remote reset to a VoIP gateway at a valid IP address. This enables the replies to get through to the VoIP gateway even if the BootRequest is blocked by the firewall. To turn off this feature, set the **Number of Timed Replies** = 0.

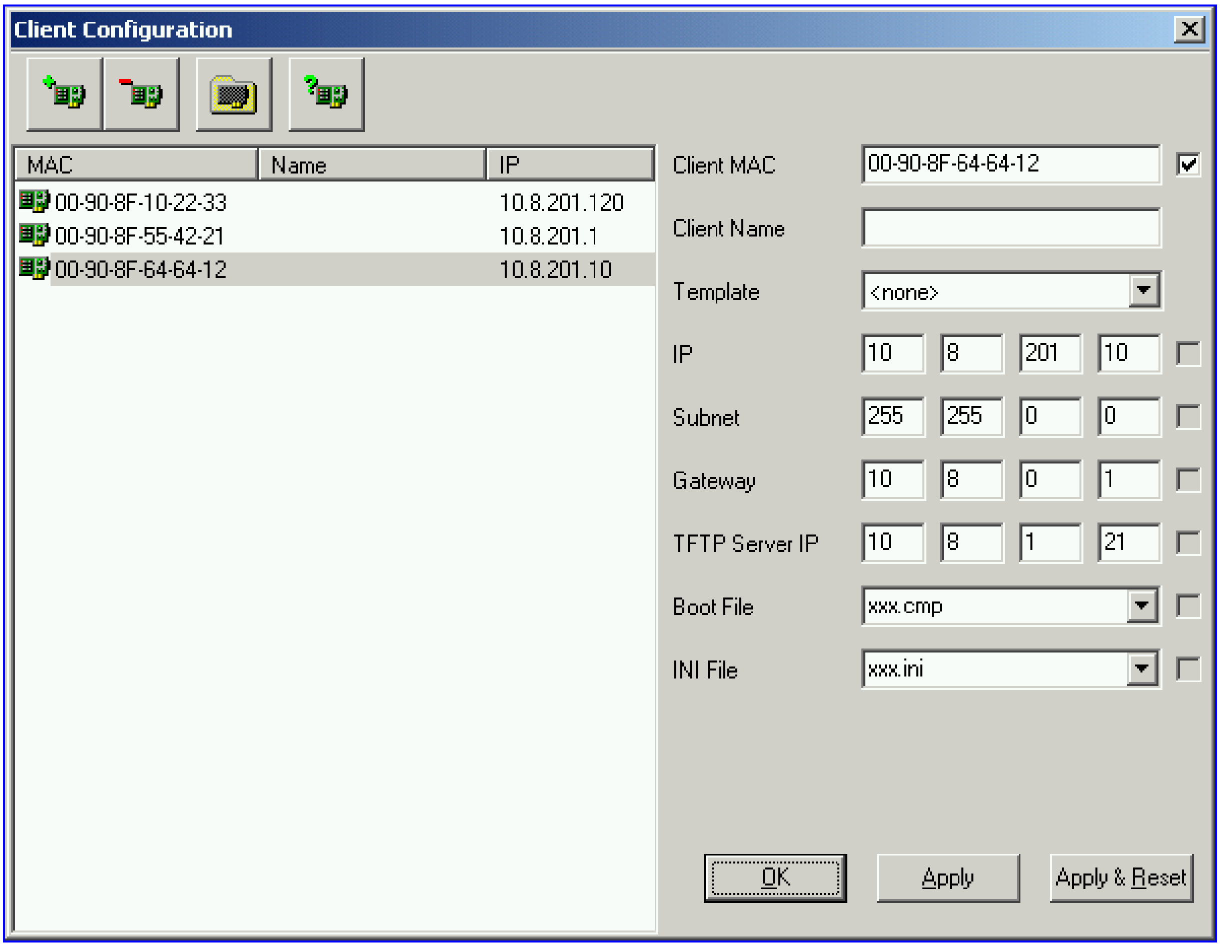
D.10.2 TFTP Preferences

- **Enabled:** To enable the TFTP functionality of the BootP Tool, select the check box. If you want to use another TFTP application, other than the one included with the BootP Tool, unselect the box.
- **On Interface:** This pull down menu displays all network interfaces currently available on the computer. Select the interface that you want to use for the TFTP. Normally, there is only one choice.
- **Directory:** This option is enabled only when the TFTP is enabled. Use this parameter to specify the folder that contains the files for the TFTP utility to manage (*cmp*, *ini*, Call Progress Tones, etc.).
- **Boot File Mask:** Boot File Mask specifies the file extension used by the TFTP utility for the boot file that is included in the BootReply message. This is the file that contains VoIP gateway software and normally appears as *cmp*.
- ***ini* File Mask:** *ini* File mask specifies the file extension used by the TFTP utility for the configuration file that is included in the BootReply message. This is the file that contains VoIP gateway configuration parameters and normally appears as *ini*.
- **Timeout:** This specifies the number of seconds that the TFTP utility waits before retransmitting TFTP messages. This can be left at the default value of 5 (the more congested your network, the higher the value you should define in these fields).
- **Maximum Retransmissions:** This specifies the number of times that the TFTP utility tries to resend messages after timing out. This can be left at the default value of 10 (the more congested your network, the higher the value you should define in these fields).

D.11 Configuring the BootP Clients

The Clients window, shown in Figure D-4 below, is used to set up the parameters for each specific VoIP gateway.

Figure D-4: Client Configuration Screen



Client Configuration

MAC	Name	IP
00-90-8F-10-22-33		10.8.201.120
00-90-8F-55-42-21		10.8.201.1
00-90-8F-64-64-12		10.8.201.10

Client MAC	00-90-8F-64-64-12				☑
Client Name					
Template	<none>				
IP	10	8	201	10	□
Subnet	255	255	0	0	□
Gateway	10	8	0	1	□
TFTP Server IP	10	8	1	21	□
Boot File	xxx.cmp				□
INI File	xxx.ini				□

D.11.1 Adding Clients

Adding a client creates an entry in the BootP Tool for a specific gateway.

➢ To add a client to the list without using a template, take these 3 steps:

1. Click the **Add New Client** icon ; a client with blank parameters is displayed.
2. Enter values in the fields on the right side of the window, using the guidelines for the fields in Section D.11.5 on page 421.
3. Click **Apply** to save this entry to the list of clients, or click **Apply & Reset** to save this entry to the list of clients and send a reset message to that gateway to immediately implement the settings.

   Note: To use **Apply & Reset** you must enable **ARP Manipulation** in the **Preferences** window. Also, you must have administrator privileges for the computer you are using.

An easy way to create several clients that use similar settings is to create a template. For information on how to create a template, refer to Section D.12 on page 424.

➢ To add a client to the list using a template, take these 5 steps:

1. Click the **Add New Client** icon ; a client with blank parameters is displayed.
2. In the field **Template**, located on the right side of the **Client Configuration Window**, click on the down arrow to the right of the entry field and select the template that you want to use.
3. The values provided by the template are automatically entered into the parameter fields on the right side of the **Client Configuration Window**. To use the template parameters, leave the check box next to that parameter selected. The parameter values appear in gray text.
4. To change a parameter to a different value, unselect the check box to the right of that parameter. This clears the parameter provided by the template and enables you to edit the entry. Clicking the check box again restores the template settings.
5. Click **Apply** to save this entry to the list of clients or click **Apply & Reset** to save this entry to the list of clients and send a reset message to that gateway to immediately implement the settings.

   Note: To use **Apply & Reset** you must enable **ARP Manipulation** in the **Preferences** window. Also, you must have administrator privileges for the computer you are using.

D.11.2 Deleting Clients

➢ To delete a client from the BootP Tool, take these 3 steps:

1. Select the client that you wish to delete by clicking on the line in the window for that client.
2. Click the **Delete Current Client** button ; a warning pops up.
3. To delete the client, click **Yes**.

D.11.3 Editing Client Parameters

➢ To edit the parameters for an existing client, take these 4 steps:

1. Select the client that you wish to edit by clicking on the line in the window for that client.
2. Parameters for that client display in the parameter fields on the right side of the window.
3. Make the changes required for each parameter.
4. Click **Apply** to save the changes, or click **Apply & Reset** to save the changes and send a reset message to that gateway to immediately implement the settings. **Note:** To use **Apply & Reset** you must enable **ARP Manipulation** in the **Preferences** window. Also, you must have administrator privileges for the computer you are using.

D.11.4 Testing the Client

There should only be one BootP utility supporting any particular client MAC active on the network at any time.

➢ To check if other BootP utilities support this client, take these 4 steps:

1. Select the client that you wish to test by clicking on the client name in the main area of the **Client Configuration Window**.
2. Click the **Test Selected Client** button

3. Examine the Log Window on the Main Application Screen. If there is another BootP utility that supports this client MAC, there is a response indicated from that utility showing the status Listed At along with the IP address of that utility.
4. If there is another utility responding to this client, you must remove that client from either this utility or the other one.

D.11.5 Setting Client Parameters

Client parameters are listed on the right side of the **Client Configuration Window**.

- **Client MAC:** The Client MAC is used by BootP to identify the VoIP gateway. The MAC address for the VoIP gateway is printed on a label located on the VoIP gateway hardware. Enter the Ethernet MAC address for the VoIP gateway in this field. Click the box to the right of this field to enable this particular client in the BootP tool (if the client is disabled, no replies are sent to BootP requests).
  Note: When the MAC address of an existing client is edited, a new client is added, with the same parameters as the previous client.
- **Client Name:** Enter a descriptive name for this client so that it is easier to remember which VoIP gateway the record refers to. For example, this name could refer to the location of the gateway.
- **Template:** Click the pull down arrow if you wish to use one of the templates that you configured. This applies the parameters from that template to the remaining fields. Parameter values that are applied by the template are indicated by a check mark in the box to the right of that parameter. Uncheck this box if you want to enter a different value. If templates are not used, the box to the right of the parameters is colored gray and is not selectable.
- **IP:** Enter the IP address you want to apply to the VoIP gateway. Use the normal dotted decimal format.
- **Subnet:** Enter the subnet mask you want to apply to the VoIP gateway. Use the normal dotted decimal format. Ensure that the subnet mask is correct. If the address is incorrect, the VoIP gateway may not function until the entry is corrected and a BootP reset is applied.
- **Gateway:** Enter the IP address for the data network gateway used on this subnet that you want to apply to the VoIP gateway. The data network gateway is a device, such as a router, that is used in the data network to interface this subnet to the rest of the enterprise network.
- **TFTP Server IP:** This field contains the IP address of the TFTP utility that is used for file transfer of software and initialization files to the gateway. When creating a new client, this field is populated with the IP address used by the BootP Tool. If a different TFTP utility is to be used, change the IP address in this field to the IP address used by the other utility.

- **Boot File:** This field specifies the file name for the software (*cmp*) file that is loaded by the TFTP utility to the VoIP gateway after the VoIP gateway receives the BootReply message. The actual software file is located in the TFTP utility directory that is specified in the BootP **Preferences** window. The software file can be followed by command line switches. For information on available command line switches, refer to Section D.11.6 on page 422.

> **Note 1:** Once the software file loads into the gateway, the gateway begins functioning from that software. In order to save this software to non-volatile memory, (only the *cmp* file, i.e., the compressed firmware file, can be burned to your device's flash memory), the -fb flag must be added to the end of the file name. If the file is not saved, the gateway reverts to the old version of software after the next reset.
>
> **Note 2:** The **Boot file** field can contain up to two file names: *cmp* file name to be used for load of application image and *ini* file name to be used for gateway provisioning. Either one, two or no file names can appear in the **Boot file** field. To use both file names use the ';' separator (without blank spaces) between the xxx.*cmp* and the yyy.*ini* files (e.g., *ram.cmp;SIPgw.ini*).

- ***ini* File:** This field specifies the configuration *ini* file that the gateway uses to program its various settings. Enter the name of the file that is loaded by the TFTP utility to the VoIP gateway after it receives the BootReply message. The actual *ini* file is located in the TFTP utility directory that is specified in the BootP Preferences window.

D.11.6 Using Command Line Switches

You can add command line switches in the field **Boot File**.

➢ To use a Command Line Switch, take these 4 steps:

1. In the field **Boot File**, leave the file name defined in the field as it is (e.g., ramxxx.*cmp*).
2. Place your cursor after *cmp.*
3. Press the space bar.
4. Type in the switch you require.

Example: 'ramxxx.*cmp* –fb' to burn flash memory.

'ramxxx.*cmp* -fb -em 4' to burn flash memory and for Ethernet Mode 4 (auto-negotiate).

Table D-1 lists and describes the switches that are available:

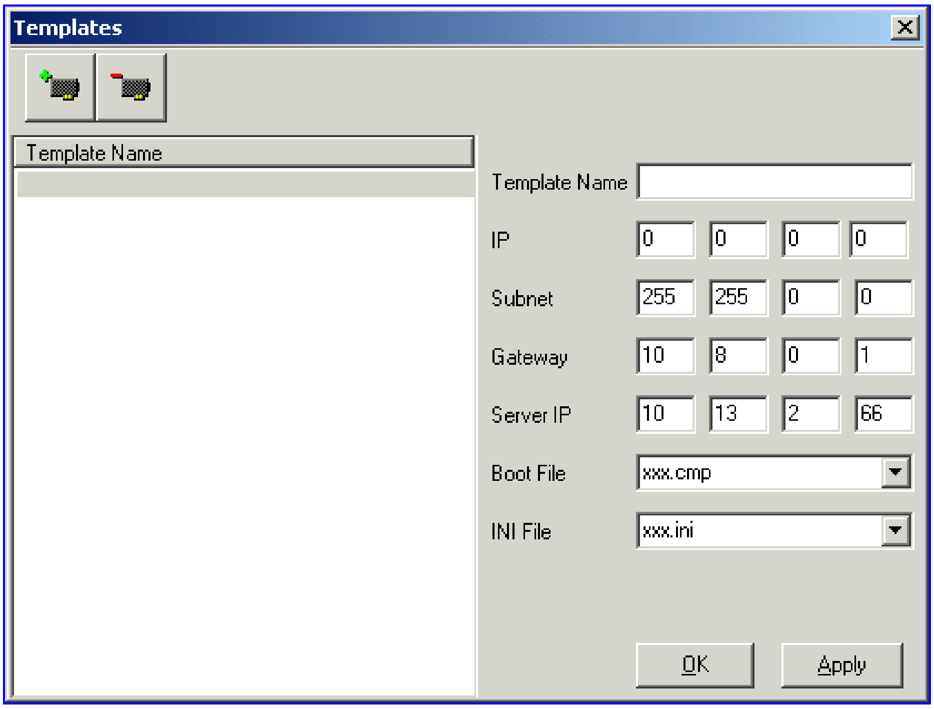
Table D-1: Command Line Switch Descriptions

<table>
<tr><th>Switch</th><th colspan="2">Description</th></tr>
<tr><td>-fb</td><td colspan="2">Burn ram.<em>cmp</em> in flash (only for <em>cmp</em> files)</td></tr>
<tr><td>-em #</td><td colspan="2">Use this switch to set Ethernet mode.
0 = 10 Base-T half-duplex
1 = 10 Base-T full-duplex
2 = 100 Base-TX half-duplex
3 = 100 Base-TX full-duplex
4 = auto-negotiate (default)
For detailed information on Ethernet interface configuration, refer to Section 10.1 on page 301.</td></tr>
<tr><td rowspan="2">-br</td><td colspan="2">This parameter is used to:
<strong>Note:</strong> This switch takes effect only from the next gateway reset.</td></tr>
<tr><td>Set the number of BootP requests the gateway sends during start-up. The gateway stops sending BootP requests when either BootP reply is received or number of retries is reached.
1 = 1 BootP retry, 1 second
2 = 2 BootP retries, 3 seconds
3 = 3 BootP retries, 6 seconds
4 = 10 BootP retries, 30 seconds
5 = 20 BootP retries, 60 seconds
6 = 40 BootP retries, 120 seconds
7 = 100 BootP retries, 300 seconds
15 = BootP retries indefinitely</td><td>Set the number of DHCP packets the gateway sends.
After all packets were sent, if there's still no reply, the gateway loads from flash.
1 = 4 DHCP packets
2 = 5 DHCP packets
3 = 6 DHCP packets (default)
4 = 7 DHCP packets
5 = 8 DHCP packets
6 = 9 DHCP packets
7 = 10 DHCP packets
15 = 18 DHCP packets</td></tr>
<tr><td>-bs</td><td colspan="2">Use –bs 1 to enable the Selective BootP mechanism.
Use –bs 0 to disable the Selective BootP mechanism.
The Selective BootP mechanism (available from Boot version 1.92) enables the gateway’s integral BootP client to filter unsolicited BootP/DHCP replies (accepts only BootP replies that contain the text ‘AUDC’ in the vendor specific information field). This option is useful in environments where enterprise BootP/DHCP servers provide undesired responses to the gateway’s BootP requests.</td></tr>
<tr><td>-be</td><td colspan="2">Use -be 1 for the device to send device-related initial startup information (such as board type, current IP address, software version) in the vendor specific information field (in the BootP request). This information can be viewed in the main screen of the BootP/TFTP, under column 'Client Info‘ (refer to Figure D-1 showing BootP/TFTP main screen with the column 'Client Info' on the extreme right). For a full list of the vendor specific Information fields, refer to Section 7.3.2 on page 261.
<strong>Note:</strong> This option is not available on DHCP servers.</td></tr>
</table>

D.12 Managing Client Templates

Templates can be used to simplify configuration of clients when most of the parameters are the same.

Figure D-5: Templates Screen



Template Name

IP: 0 . 0 . 0 . 0

Subnet: 255 . 255 . 0 . 0

Gateway: 10 . 8 . 0 . 1

Server IP: 10 . 13 . 2 . 66

Boot File: xxx.cmp

INI File: xxx.ini

➢ To create a new template, take these 4 steps:

1. Click the **Add New Template** button
2. Fill in the default parameter values in the parameter fields.
3. Click **Apply** to save this new template.
4. Click **OK** when you are finished adding templates.

➢ To edit an existing template, take these 4 steps:

1. Select the template by clicking on its name from the list of templates in the window.
2. Make changes to the parameters, as required.
3. Click **Apply** to save this new template.
4. Click **OK** when you are finished editing templates.

➢ To delete an existing template, take these 3 steps:

1. Select the template by clicking its name from the list of templates in the window.
2. Click the **Delete Current Template** button ; A warning pop up message appears.
3. To delete the template, click **Yes**.
   Note that if this template is currently in use, the template cannot be deleted.

E RTP/RTCP Payload Types and Port Allocation

RTP Payload Types are defined in RFC 3550 and RFC 3551. We have added new payload types to enable advanced use of other coder types. These types are reportedly not used by other applications.

E.1 Packet Types Defined in RFC 3551

Table E-1: Packet Types Defined in RFC 3551

Payload Type	Description	Basic Packet Rate [msec]
0	G.711 µ-Law	10,20
2	G.726-32	10,20
4	G.723 (6.3/5.3 kbps)	30
8	G.711 A-Law	10,20
18	G.729A/B	20
200	RTCP Sender Report	Randomly, approximately every 5 seconds (when packets are sent by channel)
201	RTCP Receiver Report	Randomly, approximately every 5 seconds (when channel is only receiving)
202	RTCP SDES packet	
203	RTCP BYE packet	
204	RTCP APP packet	

E.2 Defined Payload Types

Table E-2: Defined Payload Types

Payload Type	Description	Basic Packet Rate [msec]
96	RFC 2833 DTMF relay	20
102	Fax Bypass	20
103	Modem Bypass	20
104	RFC 2198 (Redundancy)	Same as channel’s voice coder.
105	NSE Bypass	

E.3 Default RTP/RTCP/T.38 Port Allocation

The following table shows the default RTP/RTCP/T.38 port allocation.

Table E-3: Default RTP/RTCP/T.38 Port Allocation

Channel Number	RTP Port	RTCP Port	T.38 Port
1	6000	6001	6002
2	6010	6011	6012
3	6020	6021	6022
4	6030	6031	6032
5	6040	6041	6042
6	6050	6051	6052
7	6060	6061	6062
8	6070	6071	6072
:	:	:	:
n	**6000 + 10(n-1)**	**6001 + 10(n-1)**	**6002 + 10(n-1)**
:	:	:	:
96	6950	6951	6952
:	:	:	:
120	7190	7191	7192

Note: To configure the gateway to use the same port for both RTP and T.38 packets, set the parameter 'T38UseRTPPort' to 1.

F RTP Control Protocol Extended Reports (RTCP-XR)

RTP Control Protocol Extended Reports (RTCP-XR) is a VoIP management control that defines a set of metrics containing information for assessing VoIP call quality and diagnosing problems. RTCP-XR (RFC 3611) extends the RTCP reports defined in RFC 3550 by providing additional VoIP metrics.

RTCP-XR information publishing is implemented in the media server according to <draft-johnston-sipping-rtcp-summary-07>. This draft defines how a SIP User Agent (UA) publishes the detailed information to a defined collector.

RTCP-XR messages containing key call-quality-related metrics are exchanged periodically (user-defined) between the gateway and the SIP UA. This allows an analyzer to monitor these metrics midstream, or a gateway to retrieve them using SNMP. The gateway can send RTCP-XR reports to an Event State Compositor (ESC) server using PUBLISH messages. These reports can be sent at the end of each call (configured using RTCPXRReportMode) and according to a user-defined interval (RTCPInterval or DisableRTCPRandomize) between consecutive reports.

To enable RTCP-XR reporting, the VQMonEnable *ini* file parameter must be set to 1.

For a detailed description of the RTCP-XR parameters, refer to 5.6.2.3 on page 154.

RTCP-XR measures VoIP call quality such as packet loss, delay, signal / noise / echo levels, estimated R-factor, and mean opinion score (MOS). RTCP-XR measures these parameters using the metrics listed in the table below.

Table F-1: RTCP-XR Published VoIP Metrics (continues on pages 427 to 428)

Metric Name
General
Start Timestamp
Stop Timestamp
Call-ID
Local Address (IP, Port & SSRC)
Remote Address (IP, Port & SSRC)
Session Description
Payload Type
Payload Description
Sample Rate
Frame Duration
Frame Octets
Frames per Packets
Packet Loss Concealment
Silence Suppression State
Jitter Buffer
Jitter Buffer Adaptive
Jitter Buffer Rate
Jitter Buffer Nominal
Jitter Buffer Max
Jitter Buffer Abs Max
Packet Loss
Network Packet Loss Rate
Jitter Buffer Discard Rate

Table F-1: RTCP-XR Published VoIP Metrics (continues on pages 427 to 428)

Metric Name
Burst Gap Loss
Burst Loss Density
Burst Duration
Gap Loss Density
Gap Duration
Minimum Gap Threshold
Delay
Round Trip Delay
End System Delay
One Way Delay
Interarrival Jitter
Min Absolute Jitter
Signal
Signal Level
Noise Level
Residual Echo Return Noise
Quality Estimates
Listening Quality R
RLQ Est. Algorithm
Conversational Quality R
RCQ Est. Algorithm
External R In
Ext. R In Est. Algorithm
External R Out
Ext. R Out Est. Algorithm
MOS-LQ
MOS-LQ Est. Algorithm
MOS-CQ
MOS-CQ Est. Algorithm
QoE Est. Algorithm

G Accessory Programs and Tools

The accessory applications and tools shipped with the device provide you with friendly interfaces that enhance device usability and smooth your transition to the new VoIP infrastructure. The following applications are available:

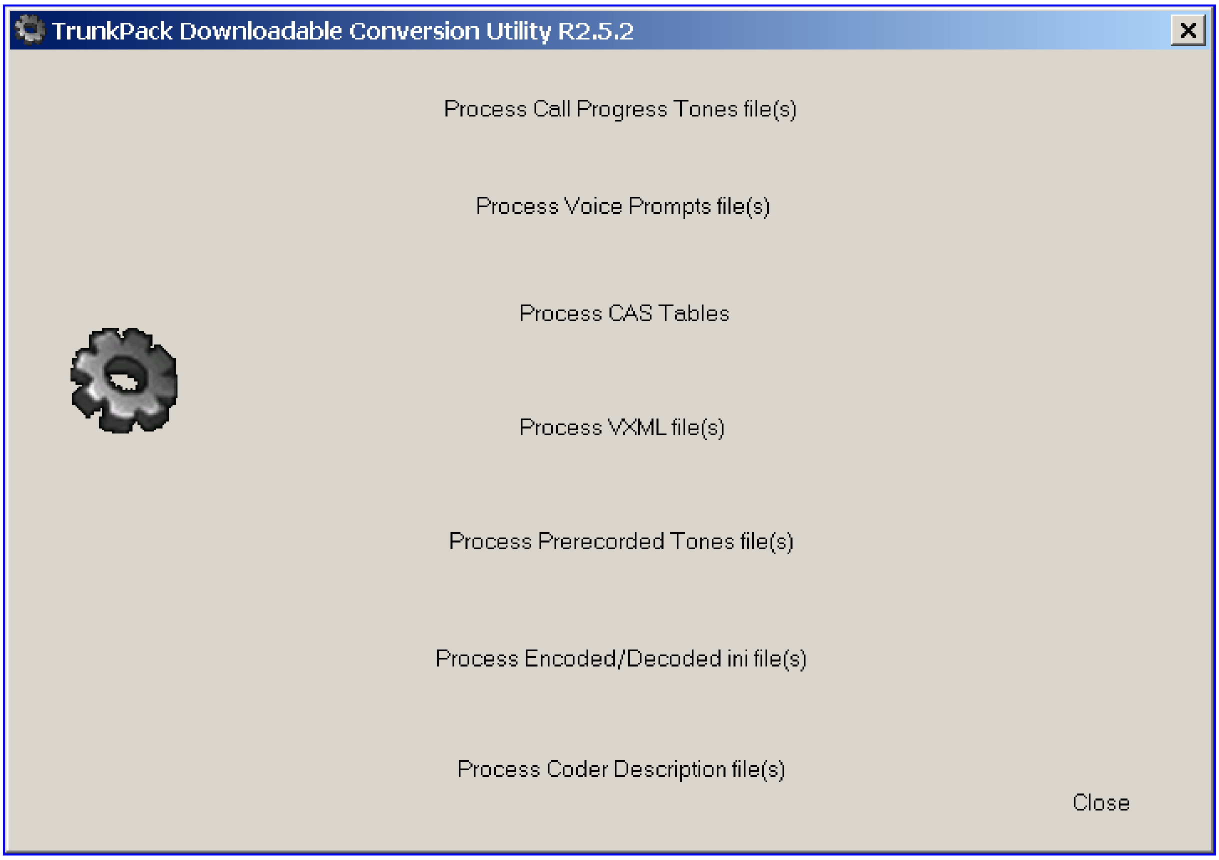
- TrunkPack Downloadable Conversion Utility (refer to Section G.1 below).
- Call Progress Tones Wizard (refer to Section G.2 on page 435).
- PSTN Trace Utility (refer to Section G.3 on page 442).

G.1 TrunkPack Downloadable Conversion Utility

Use the TrunkPack Downloadable Conversion Utility to:

- Create a loadable Call Progress Tones file (refer to Section G.1.1 on page 430).
- To create a loadable CAS protocol table file (refer to Section G.1.2 on page 431).
- Encode / decode an *ini* file (refer to Section G.1.3 on page 432)
- Create a loadable Prerecorded Tones file (refer to Section G.1.4 on page 433).

Figure G-1: TrunkPack Downloadable Conversion Utility Opening Screen


TrunkPack Downloadable Conversion Utility R2.5.2
Process Call Progress Tones file(s)
Process Voice Prompts file(s)
Process CAS Tables
Process VXML file(s)
Process Prerecorded Tones file(s)
Process Encoded/Decoded ini file(s)
Process Coder Description file(s)
Close

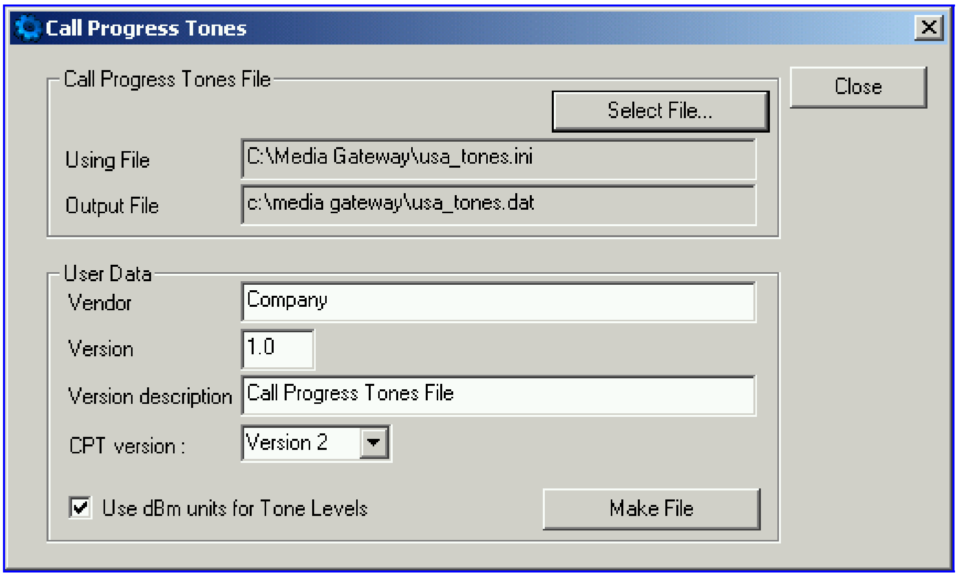
G.1.1 Converting a CPT *ini* File to a Binary *dat* File

For detailed information on creating a CPT *ini* file, refer to Section 17.1 on page 391.

➢ To convert a CPT *ini* file to a binary *dat* file, take these 10 steps:

1. Execute the TrunkPack Downloadable Conversion Utility, DConvert.exe (supplied with the software package); the utility's main screen opens (shown in Figure G-1).
2. Click the **Process Call Progress Tones File(s)** button; the 'Call Progress Tones' screen, shown in Figure G-2, opens.

Figure G-2: Call Progress Tones Conversion Screen



Call Progress Tones

Call Progress Tones File

Using File: C:\Media Gateway\usa_tones.ini

Output File: c:\media gateway\usa_tones.dat

User Data

Vendor: Company

Version: 1.0

Version description: Call Progress Tones File

CPT version : Version 2

☑ Use dBm units for Tone Levels

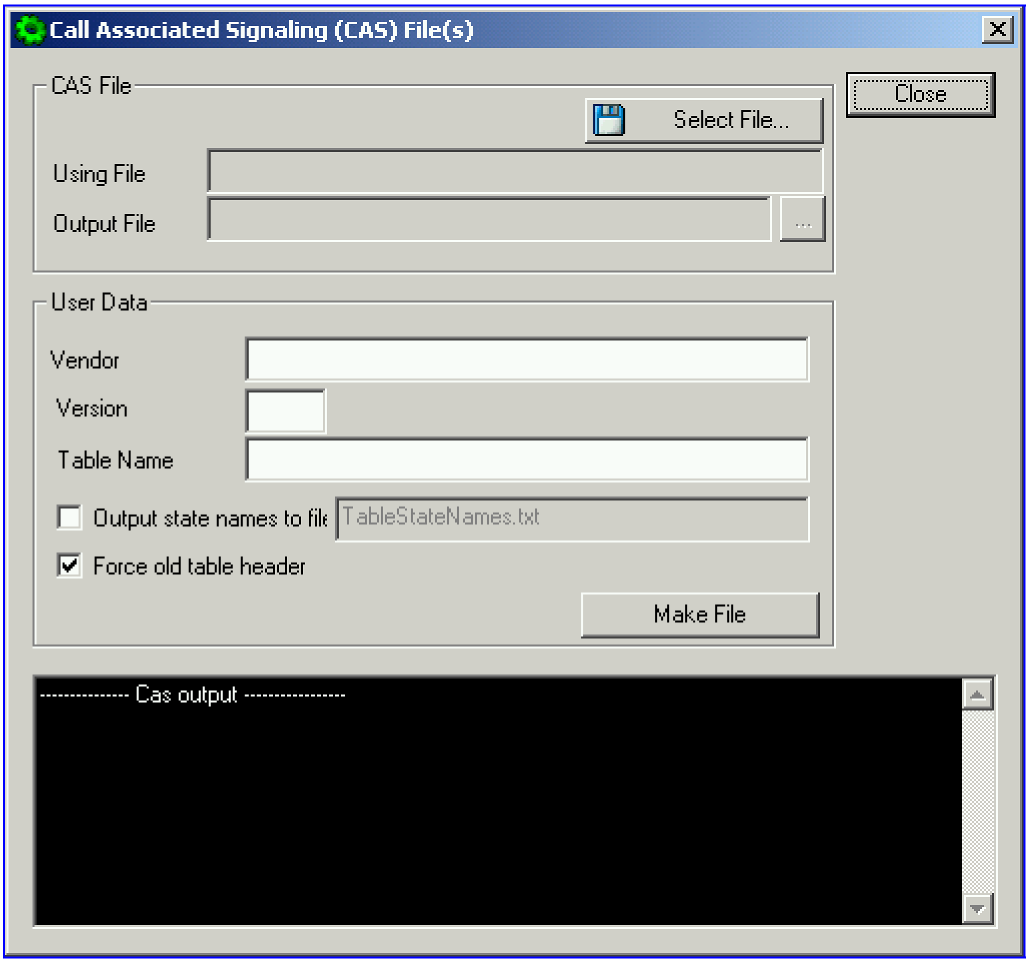
3. Click the **Select File…** button that is in the 'Call Progress Tone File' box.
4. Navigate to the folder that contains the CPT *ini* file you want to convert.
5. Click the *ini* file and click the **Open** button; the name and path of both the *ini* file and the (output) *dat* file appears in the fields below the Select File button.
6. Enter the Vendor Name, Version Number and Version Description in the corresponding required fields under the 'User Data' section.
   - The maximum length of the Vendor field is 256 characters.
   - The format of the Version field is composed of two integers separated by a period '.' (e.g., 1.2, 23.4, 5.22).
   - The maximum length of the Version Description field is 256 characters.
7. The default value of the CPT Version drop-down list is Version 3. Do one of the following:
   - If the software version you are using is prior to version 4.4, select Version 1 (to maintain backward compatibility).
   - If the software version you are using is 4.4, select Version 2.
   - Otherwise, leave the value at its default.
8. Check the 'Use dBm units for Tone Levels' check box. Note that the levels of the Call Progress Tones (in the CPT file) must be in -dBm units.
9. Click the **Make File** button; you're prompted that the operation (conversion) was successful.
10. Close the application.

G.1.2 Creating a loadable CAS Protocol Table File

➢ To create a loadable CAS protocol table file take the next 11 steps:

1. Construct the CAS protocol files (xxx.txt and UserProt_defines_xxx.h).
2. Copy the files generated in the previous step to the same directory the TrunkPack Downloadable Conversion utility is located and ensure that the files CASSetup.h and cpp.exe are also located in the same directory.
3. Execute the TrunkPack Downloadable Conversion utility, DConvert.exe (supplied with the software package); the utility's main screen opens (shown in Figure G-1).
4. Click the button **Process CAS Tables**; the Call Associated Signaling (CAS) screen, shown in Figure G-3 below, opens.

Figure G-3: Call Associated Signaling (CAS) Screen



5. Click the button **Select File…** under the section 'CAS File'; a Browse window appears.
6. Navigate to the desired location and select the *txt* file you want to converted; this automatically designates the output file as the same name and path, but with a *dat* extension, the table's name is also automatically designated.
7. Enter the vendor name and version number in the required fields under the section 'User Data':
   - **Vendor** - maximum of 32 characters.
   - **Version** - must be in the format: [number] [single period '.'] [number] (e.g., 1.2, 23.4, 5.22).
8. Modify the **Table Name** according to your requirements.

9. To create a file (for troubleshooting purposes) that contains the name of the States and their actual values:

   Check the **Output state names to file** checkbox; the default file name *TableStateNames.txt* appears in the adjacent field (you can modify the name of the file). The generated file is to be located in the same directory as the TrunkPack Downloadable Conversion utility.

10. Uncheck the **Force old table header** checkbox.

11. Click the button **Make File**; the *dat* file is generated and placed in the directory specified in the field **Output File**. A message box informing you that the operation was successful indicates that the process is completed. On the bottom of the Call Assisted Signaling (CAS) Files(s) screen, the CAS output log box displays the log generated by the process. It can be copied as needed. The information in it isn't retained after the screen is closed.

G.1.3 Encoding / Decoding an *ini* File

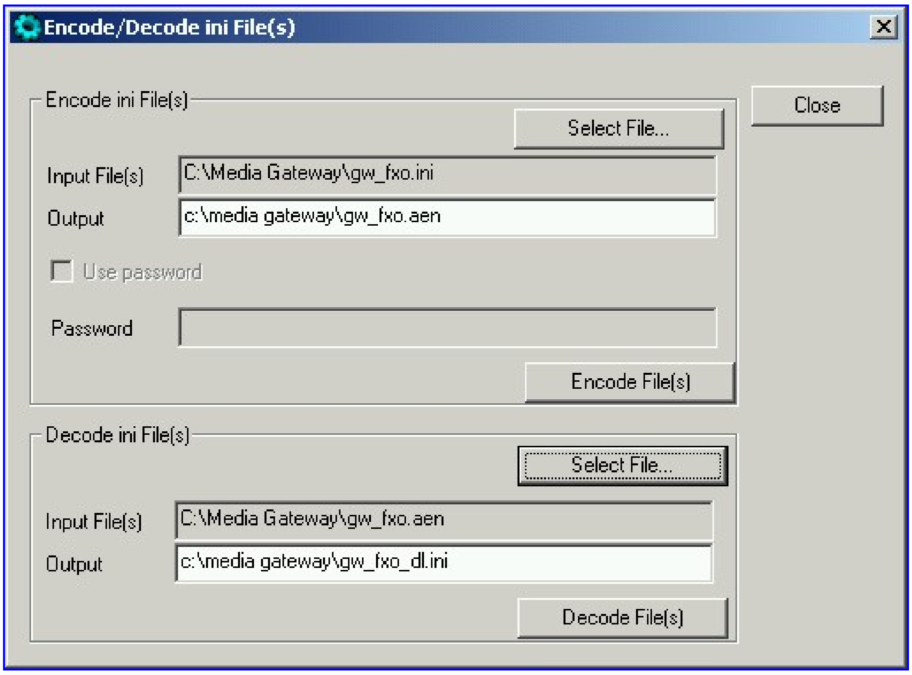
For detailed information on secured *ini* file, refer to Section 6.1 on page 213.

➢ To encode an *ini* file, take these 6 steps:

1. Execute the TrunkPack Downloadable Conversion Utility, DConvert.exe (supplied with the software package); the utility's main screen opens (shown in Figure G-1).

2. Click the **Process Encoded/Decoded *ini* file(s)** button; the 'Encode/Decode *ini* File(s)' screen, shown in Figure G-4, opens.

Figure G-4: Encode/Decode *ini* File(s) Screen



Encode/Decode ini File(s)

Encode ini File(s)

Input File(s): C:\Media Gateway\gw_fxo.ini

Output: c:\media gateway\gw_fxo.aen

□ Use password

Password:

Decode ini File(s)

Input File(s): C:\Media Gateway\gw_fxo.aen

Output: c:\media gateway\gw_fxo_dl.ini

3. Click the **Select File…** button under the 'Encode *ini* File(s)' section.

4. Navigate to the folder that contains the *ini* file you want to encode.

5. Click the *ini* file and click the **Open** button; the name and path of both the *ini* file and the output encoded file appear in the fields under the **Select File** button. Note that the name and extension of the output file can be modified.

6. Click the **Encode File(s)** button; an encoded *ini* file with the name and extension you specified is created.

➢ To decode an encoded *ini* file, take these 4 steps:

1. Click the **Select File…** button under the 'Decode *ini* File(s)' section.
2. Navigate to the folder that contains the file you want to decode.
3. Click the file and click the **Open** button. the name and path of both the encode *ini* file and the output decoded file appear in the fields under the **Select File** button. Note that the name of the output file can be modified.
4. Click the **Decode File(s)** button; a decoded *ini* file with the name you specified is created.

> **Note:** The decoding process verifies the input file for validity. Any change made to the encoded file causes an error and the decoding process is aborted.

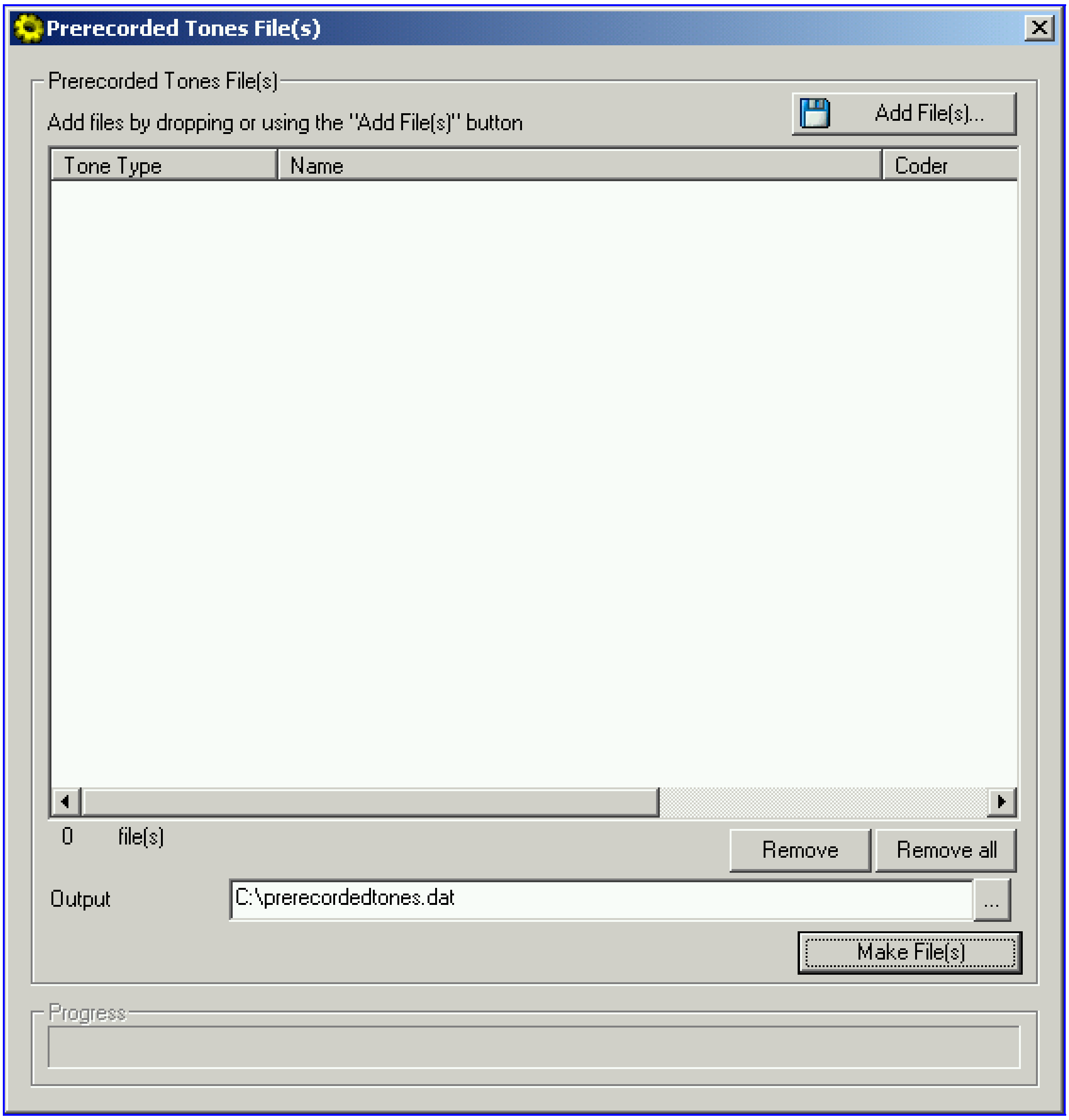
G.1.4 Creating a Loadable Prerecorded Tones File

For detailed information on the PRT file, refer to Section 17.2 on page 395.

➢ To create a loadable PRT *dat* file from your raw data files, take these 7 steps:

1. Prepare the prerecorded tones (raw data PCM or L8) files you want to combine into a single *dat* file using standard recording utilities.
2. Execute the TrunkPack Downloadable Conversion utility, DConvert.exe (supplied with the software package); the utility's main screen opens (shown in Figure G-1).
3. Click the **Process Prerecorded Tones File(s)** button; the Prerecorded Tones File(s) screen, shown in Figure G-5, opens.

Figure G-5: Prerecorded Tones Screen



Prerecorded Tones File(s)

Prerecorded Tones File(s)

Add files by dropping or using the "Add File(s)" button

Tone Type	Name	Coder

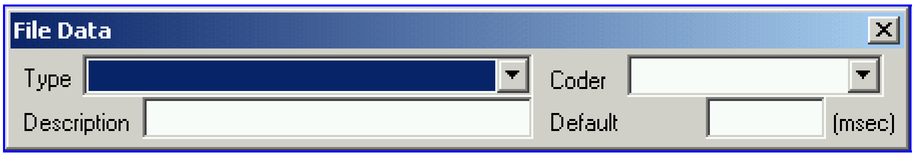
0 file(s)

Output: C:\prerecordedtones.dat

4. To add the prerecorded tone files (you created in Step 1) to the 'Prerecorded Tones' screen follow one of these procedures:
   - Select the files and drag them to the 'Prerecorded Tones' screen.
   - Click the **Add File(s)** button; the 'Select Files' screen opens. Select the required Prerecorded Tone files and click the **Add>>** button. Close the 'Select Files' screen.
5. For each raw data file, define a Tone Type, a Coder and a Default Duration by completing the following steps:
   - Double-click or right-click the required file; the 'File Data' window (shown in Figure G-6) appears.
   - From the 'Type' drop-down list, select the tone type this raw data file is associated with.
   - From the 'Coder' drop-down list, select the coder that corresponds to the coder this raw data file was *originally* recorded with.
   - In the 'Description' field, enter additional identifying information (optional).

- In the 'Default' field, enter the default duration this raw data file is repeatedly played.
- Close the 'File Data' window (press the **Esc** key to cancel your changes); you are returned to the Prerecorded Tones File(s) screen.

Figure G-6: File Data Window



6. In the 'Output' field, specify the output directory in which the PRT file is generated followed by the name of the PRT file (the default name is *prerecordedtones.dat*). Alternatively, use the Browse button to select a different output file. Navigate to the desired file and select it; the selected file name and its path appear in the 'Output' field.
7. Click the **Make File(s)** button; the Progress bar at the bottom of the window is activated. The *dat* file is generated and placed in the directory specified in the 'Output' field. A message box informing you that the operation was successful indicates that the process is completed.

G.2 Call Progress Tones Wizard

This section describes the Call Progress Tones Wizard (CPTWizard), an application designed to facilitate the provisioning of a Mediant 1000/FXO module by recording and analyzing Call Progress Tones generated by any PBX or telephone network.

G.2.1 About the Call Progress Tones Wizard

The Call Progress Tones wizard helps detect the Call Progress Tones generated by your PBX (or telephone exchange) and creates a basic Call Progress Tones *ini* file (containing definitions for all relevant Call Progress Tones), providing a good starting point when configuring an Mediant 1000/FXO module. This *ini* file can then be converted to a *dat* file that can be loaded to the gateway using the TrunkPack Downloadable Conversion utility.

To use this wizard, a Mediant 1000/FXO module connected to your PBX with 2 physical phone lines is required. This gateway must be configured with factory-default settings and shouldn't be used for phone calls during the operation of the wizard.

Note that firmware version 4.2 and above is required on the gateway.

G.2.2 Installation

The CPTWizard can be installed on any Windows 2000 or Windows XP based PC. Windows-compliant networking and audio peripherals are required for full functionality.

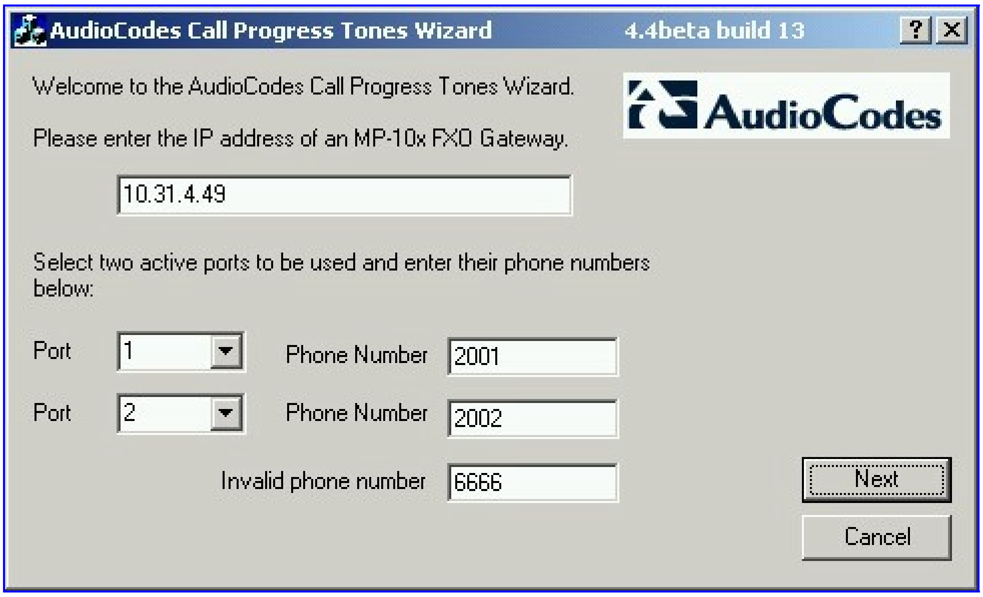
To install the CPTWizard, copy the files from the supplied installation kit to any folder on your PC. No further setup is required (approximately 5 MB of hard disk space are required).

G.2.3 Initial Settings

➢ To start the CPTWizard, take these 5 steps:

1. Execute the CPTWizard.exe file; the wizard's initial settings screen is displayed.

Figure G-7: Initial Settings Screen



AudioCodes Call Progress Tones Wizard 4.4beta build 13

Welcome to the AudioCodes Call Progress Tones Wizard.

Please enter the IP address of an MP-10x FXO Gateway.

10.31.4.49

Select two active ports to be used and enter their phone numbers below:

Port 1 Phone Number 2001

Port 2 Phone Number 2002

Invalid phone number 6666

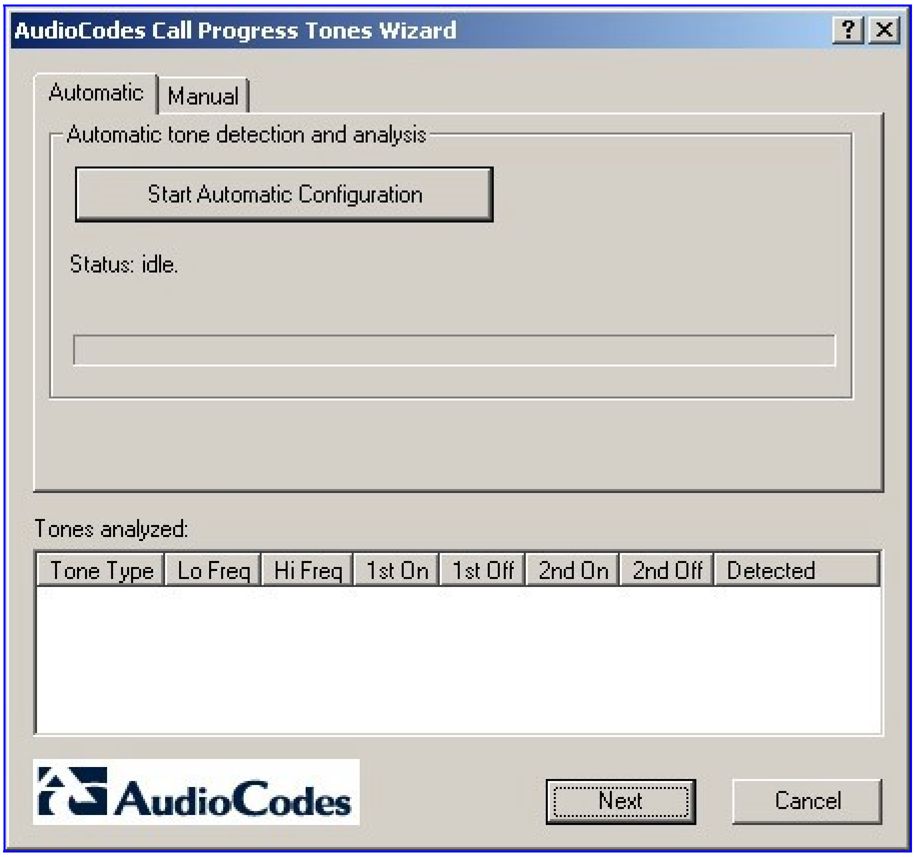
2. Enter the IP address of the Mediant 1000/FXO module you are using.
3. Select the gateway's ports that are connected to your PBX, and specify the phone number of each extension.
4. In the **Invalid phone number** field, enter a number that generates a 'fast busy' tone when dialed. Usually, any incorrect phone number should cause a 'fast busy' tone.
5. Click **Next**.

Note: The CPTWizard communicates with the FXO module via TPNCP (TrunkPack Network Control Protocol). If this protocol has been disabled in the gateway configuration, the CPTWizard doesn't display the next screen and an error is reported.

G.2.4 Recording Screen – Automatic Mode

After the connection to the Mediant 1000/FXO module is established, the recording screen is displayed.

Figure G-8: Recording Screen –Automatic Mode



AudioCodes Call Progress Tones Wizard

Automatic | Manual

Automatic tone detection and analysis

Start Automatic Configuration

Status: idle.

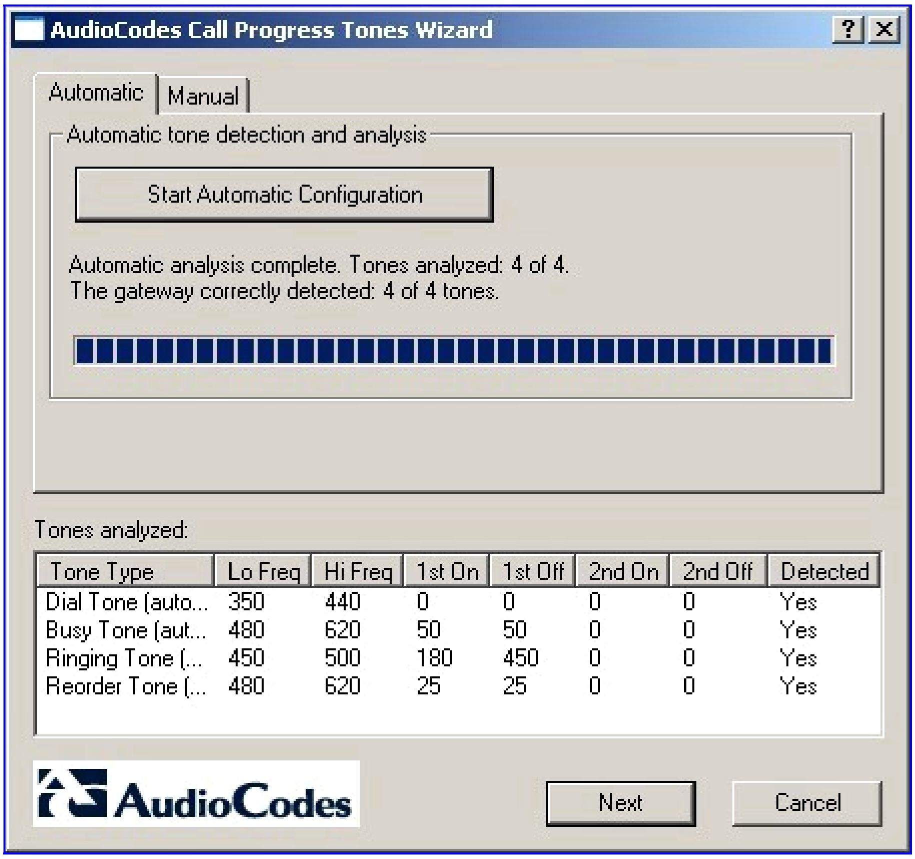
Tones analyzed:

Tone Type	Lo Freq	Hi Freq	1st On	1st Off	2nd On	2nd Off	Detected

➢ To start recording in automatic mode, take these 3 steps:

1. Click the **Start Automatic Configuration** button; the wizard starts the following Call Progress Tones detection sequence (the operation takes approximately 60 seconds to complete):
   a. Sets port 1 offhook, listens to the dial tone.
   b. Sets port 1 and port 2 offhook, dials the number of port 2, listens to the busy tone.
   c. Sets port 1 offhook, dials the number of port 2, listens to the Ringback tone.
   d. Sets port 1 offhook, dials an invalid number, listens to the reorder tone.
2. The wizard then analyzes the recorded Call Progress Tones and displays a message specifying the tones that were detected (by the gateway) and analyzed (by the wizard) correctly. At the end of a successful detection operation, the detected Call Progress Tones are displayed in the **Tones Analyzed** pane (refer to Figure G-9).

Figure G-9: Recording Screen after Automatic Detection



AudioCodes Call Progress Tones Wizard

Automatic | Manual

Automatic tone detection and analysis

Start Automatic Configuration

Automatic analysis complete. Tones analyzed: 4 of 4.
The gateway correctly detected: 4 of 4 tones.

Tones analyzed:

Tone Type	Lo Freq	Hi Freq	1st On	1st Off	2nd On	2nd Off	Detected
Dial Tone (auto...	350	440	0	0	0	0	Yes
Busy Tone (aut...	480	620	50	50	0	0	Yes
Ringing Tone (...	450	500	180	450	0	0	Yes
Reorder Tone (...	480	620	25	25	0	0	Yes

3. All four Call Progress Tones are saved (as standard A-law PCM at 8000 bits per sample) in the same directory as the CPTWizard.exe file is located, with the following names:
   - cpt_recorded_dialtone.pcm
   - cpt_recorded_busytone.pcm
   - cpt_recorded_ringtone.pcm
   - cpt_recorded_invalidtone.pcm

Notes:

- If the gateway is configured correctly (with a Call Progress Tones *dat* file loaded to the gateway), all four Call Progress Tones are detected by the gateway. By noting whether the gateway detects the tones or not, you can determine how well the Call Progress Tones *dat* file matches your PBX. During the first run of the CPTWizard, it is likely that the gateway does not detect any tones.
- Some tones cannot be detected by the Mediant 1000 gateway hardware (such as 3-frequency tones and complex cadences). CPTWizard is therefore limited to detecting only those tones that can be detected on the Mediant 1000 gateway.

At this stage, you can either click **Next** to generate a Call Progress Tones *ini* file and terminate the wizard, or continue to manual recording mode.

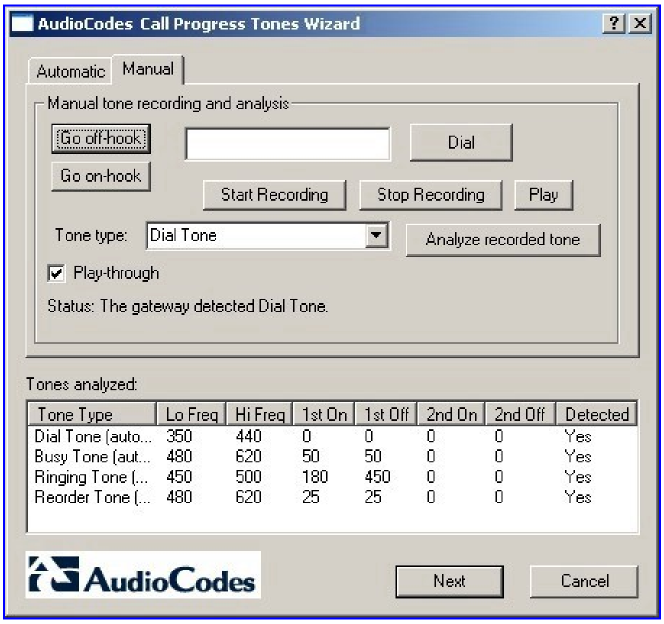
G.2.5 Recording Screen – Manual Mode

In manual mode you can record and analyze tones, included in the Call Progress Tones *ini* file, in addition to those tones analyzed when in automatic mode.

➢ To start recording in manual mode, take these 6 steps:

1. Click the **Manual** tab at the top of the recording screen, the manual recording screen is displayed.

Figure G-10: Recording Screen - Manual Mode



Tone type: Dial Tone

☑ Play-through

Status: The gateway detected Dial Tone.

Tone Type	Lo Freq	Hi Freq	1st On	1st Off	2nd On	2nd Off	Detected
Dial Tone (auto...	350	440	0	0	0	0	Yes
Busy Tone (aut...	480	620	50	50	0	0	Yes
Ringing Tone (...	450	500	180	450	0	0	Yes
Reorder Tone (...	480	620	25	25	0	0	Yes

2. Check the **play-through** check box to hear the tones through your PC speakers.
3. Click the **Go offhook** button, enter a number to dial in the **Dial** field, and click the **Dial** button. When you're ready to record, click the **Start Recording** button; when the desired tone is complete, click **Stop Recording**. (The recorded tone is saved as 'cpt_manual_tone.pcm'.)

Note: Due to some PC audio hardware limitations, you may hear 'clicks' in play-through mode. It is safe to ignore these clicks.

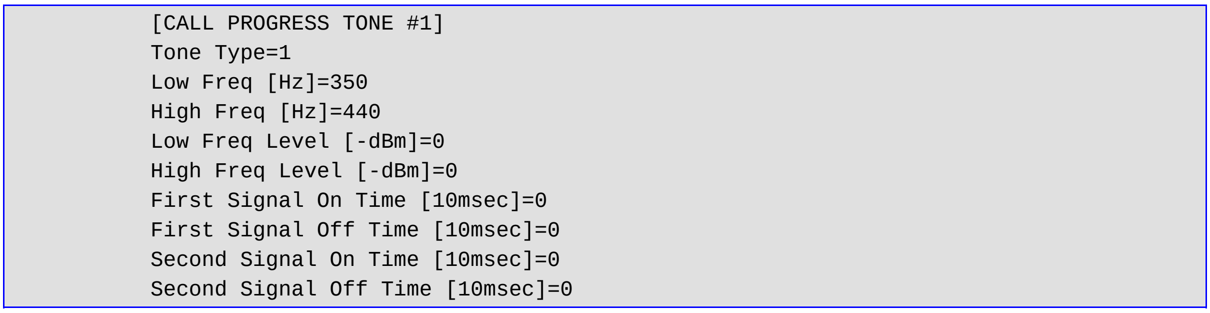
4. Select the tone type from the drop-down list and click **Analyze recorded tone**; the analyzed tone is added to the **Tones analyzed** list at the bottom of the screen. It is possible to record and analyze several different tones for the same tone type (e.g., different types of 'busy' signal).
5. Repeat the process for more tones, as necessary.
6. When you're finished adding tones to the list, click **Next** to generate a Call Progress Tones *ini* file and terminate the wizard.

G.2.6 The Call Progress Tones *ini* File

After the Call Progress Tones detection is complete, a text file named call_progress_tones.ini is created in the same directory as the directory in which the CPTWizard.exe is located. This file contains:

- Information about each tone that was recorded and analyzed by the wizard. This information includes frequencies and cadence (on/off) times, and is required for using this file with the TrunkPack Downloadable Conversion utility.

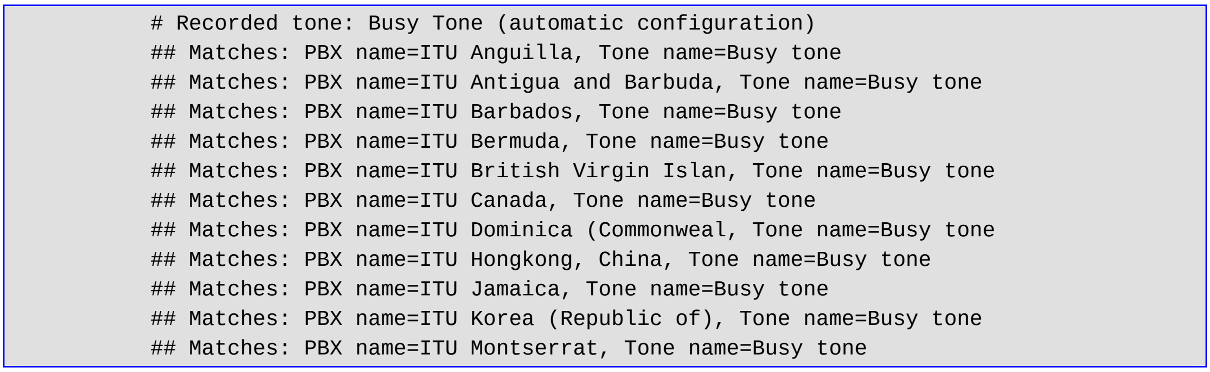
Figure G-11: Call Progress Tone Properties



```
[CALL PROGRESS TONE #1]
Tone Type=1
Low Freq [Hz]=350
High Freq [Hz]=440
Low Freq Level [-dBm]=0
High Freq Level [-dBm]=0
First Signal On Time [10msec]=0
First Signal Off Time [10msec]=0
Second Signal On Time [10msec]=0
Second Signal Off Time [10msec]=0
```

- Information related to possible matches of *each* tone with the CPTWizard's internal database of well-known tones. This information is specified as comments in the file, and is ignored by the TrunkPack Downloadable Conversion utility.

Figure G-12: Call Progress Tone Database Matches



```
# Recorded tone: Busy Tone (automatic configuration)
## Matches: PBX name=ITU Anguilla, Tone name=Busy tone
## Matches: PBX name=ITU Antigua and Barbuda, Tone name=Busy tone
## Matches: PBX name=ITU Barbados, Tone name=Busy tone
## Matches: PBX name=ITU Bermuda, Tone name=Busy tone
## Matches: PBX name=ITU British Virgin Islan, Tone name=Busy tone
## Matches: PBX name=ITU Canada, Tone name=Busy tone
## Matches: PBX name=ITU Dominica (Commonweal, Tone name=Busy tone
## Matches: PBX name=ITU Hongkong, China, Tone name=Busy tone
## Matches: PBX name=ITU Jamaica, Tone name=Busy tone
## Matches: PBX name=ITU Korea (Republic of), Tone name=Busy tone
## Matches: PBX name=ITU Montserrat, Tone name=Busy tone
```

- Information related to matches of *all* tones recorded with the CPTWizard's internal database. The database is scanned to find one or more PBX definitions that match all recorded tones (i.e., dial tone, busy tone, ringing tone, reorder tone and any other manually-recorded tone – all match the definitions of the PBX). If a match is found, the entire PBX definition is reported (as comments) in the *ini* file using the same format.

Figure G-13: Full PBX/Country Database Match

```
## Some tones matched PBX/country Audc US
## Additional database tones guessed below (remove #'s to use).
#
# # Audc US, US Ringback tone
# [CALL PROGRESS TONE #5]
# Tone Type=2
# Low Freq [Hz]=450
# High Freq [Hz]=500
# Low Freq Level [-dBm]=0
# High Freq Level [-dBm]=0
# First Signal On Time [10msec]=180
# First Signal Off Time [10msec]=450
# Second Signal On Time [10msec]=0
# Second Signal Off Time [10msec]=0
```

Note 1: If a match is found in the database, consider using the database's definitions instead of the recorded definitions, as they might be more accurate.

Note 2: For full operability of the Mediant 1000/FXO module, it may be necessary to edit this file and add more Call Progress Tone definitions. Sample Call Progress Tones *ini* files are available in the release package.

Note 3: When the CPT *ini* file is complete, use the TrunkPack Downloadable Conversion utility to create a loadable CPT *dat* file. After loading this file to the gateway, repeat the automatic detection procedure discussed above, and verify that the gateway detects all four Call Progress Tones correctly.

G.2.7 Adding a Reorder Tone to the CPT File

The following procedure describes how to add a Reorder tone that a PBX generates to indicate a disconnected call, to the CPT file.

➢ To add a Reorder tone to the CPT file, take these 11 steps:

1. Make a call (using G.711) between the Mediant 1000 FXO, which is connected to the PBX, and a remote entity in the IP network.
2. Capture the call using a network sniffer such as Wireshark.
3. Disconnect the call from the PBX side, and then wait approximately 30 seconds before stopping the Wireshark recording.
4. In the network trace, locate the RTP stream sent from the FXO.
5. Save the RTP payload on your PC as a *.pcm file by clicking **Save Payload** (**Statistics** menu > **RTP** > **Stream Analysis**. (Note: ensure that you select the 'forward' option.)
6. Open the *.pcm file in a voice recording utility such as CoolEdit.
7. Locate the tone that the PBX played to indicate the disconnected call (if such a tone exists).
8. Locate the attributes of the tone -- its frequency and interval (on / off time).

9. In the Call Progress Tones file, add a new Reorder Tone with the attributes you found in the previous step. Ensure that you update the numbers of the successive tones and the total number of tones in the beginning of the file.
10. Create a Call Progress Tones.dat file using the DConvert Utility (refer to Section G.1 on page 429).
11. Load the new file to the gateway, and then reset the gateway.

G.3 PSTN Trace Utility

These utilities are designed to convert PSTN trace binary files to textual form. The binary PSTN trace files are generated when the user sets the PSTN interface to trace mode.

G.3.1 Operation

- Generating textual trace/audit file for CAS protocols:

  To generate a readable text file out of the binary trace file when using CAS protocols, rename the PSTN trace binary file to CASTrace0.dat and copy it to the same directory in which the translation utility CAS_Trace.exe is located. Then, run CAS_Trace.exe (no arguments are required). As a result, the textual file CASTrace0.txt is created.

- Generating textual trace/audit file for ISDN PRI protocols:

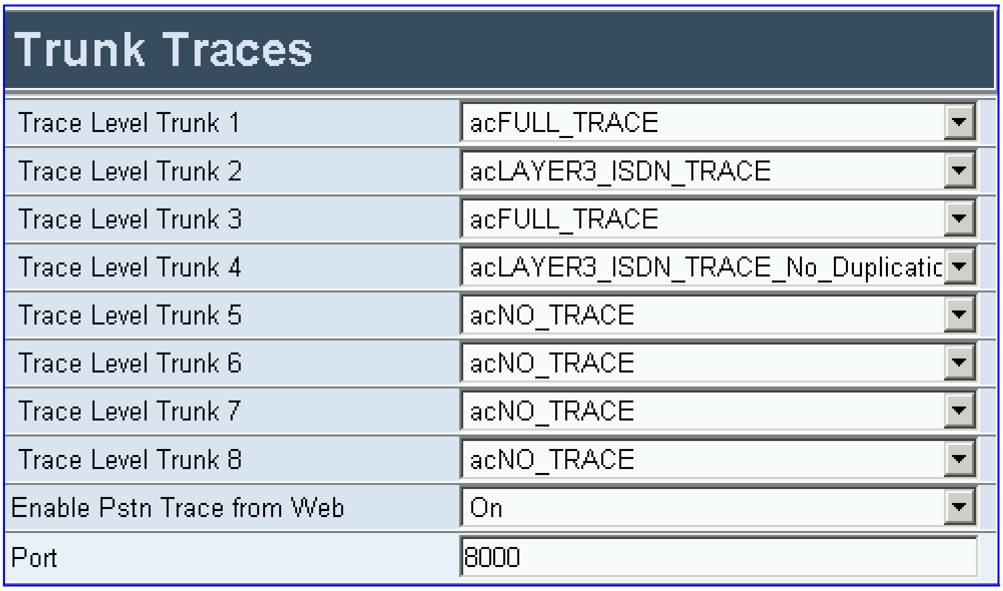
  To generate a readable text file out of the binary trace file when using ISDN protocols, copy the PSTN trace binary file to the same directory in which the translation utility **Convert_Trace.bat** is located. The following files should reside in the same directory: **Dumpview.exe, Dumpview.cfg** and **ReadMe.txt**. Please read carefully the **ReadMe.txt** in order to understand the usage of the translation utility. Next, run the **Convert_Trace.bat**. As a result, the textual file is created.

To start and collect the PSTN trace via the Web, please use the following instructions. (Refer to Figure G-14 for a view of the Trunk Traces). Also, please note if the PSTN trace was of a PRI of CAS collection based on the framer involved in the trace. This information is needed to properly parse the captured data.

➢ To start and collect the PSTN trace via the Web, take these 10 steps:

1. Run the UDP2File utility.
2. Determine the trace file name.
3. Determine the UDP port.
4. Mark the PSTN Trace check box.
5. Click the Run button=> the UDP2File utility starts to collect the trace messages.
6. Activate the Web page by entering <Mediant 1000 IP address>/TrunkTraces, such as: http://10.8.8.101/TrunkTraces. The user and password is the same for the unit.
7. In the Web page set the trace level of each trunk.
8. Enable the trace via the Web.
9. Determine the UDP port (the same as in step 3).
10. Click the Submit button => the board starts to send the trace messages. In the UDP2File utility (Refer to Figure G-15) you should see the number in the packets counter increasing.

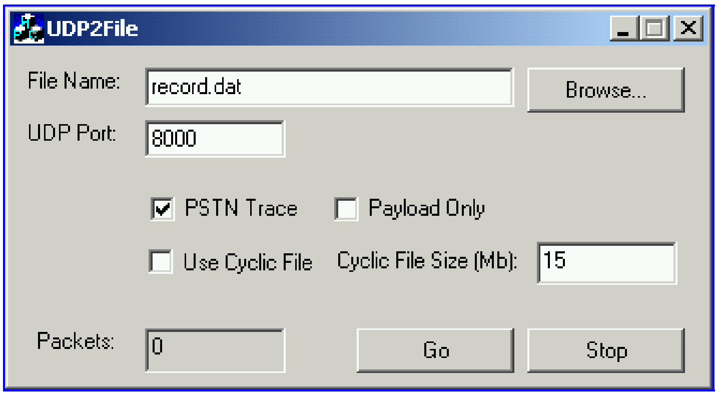
Figure G-14: Trunk Traces



Trunk Traces

Trace Level Trunk 1	acFULL_TRACE
Trace Level Trunk 2	acLAYER3_ISDN_TRACE
Trace Level Trunk 3	acFULL_TRACE
Trace Level Trunk 4	acLAYER3_ISDN_TRACE_No_Duplicatic
Trace Level Trunk 5	acNO_TRACE
Trace Level Trunk 6	acNO_TRACE
Trace Level Trunk 7	acNO_TRACE
Trace Level Trunk 8	acNO_TRACE
Enable Pstn Trace from Web	On
Port	8000

Figure G-15: UDP2File Utility



UDP2File

File Name: record.dat

UDP Port: 8000

☑ PSTN Trace ☐ Payload Only

☐ Use Cyclic File Cyclic File Size (Mb): 15

Packets: 0

Reader's Notes

H Release Reason Mapping

This appendix describes the available mapping mechanisms of SIP Responses to Q.850 Release Causes and vice versa.

Table H-1 and Table H-2 describe the existing mapping of ISDN Release Causes to SIP Responses. To override this hard-coded mapping and flexibly map SIP Responses to ISDN Release Causes, use the parameters CauseMapISDN2SIP and CauseMapSIP2ISDN (described in Section 6.5.9 on page 237), or via the Embedded Web Server (refer to Section 5.5.4.7 on page 114).

It is also possible to map the less commonly-used SIP Responses to a single default ISDN Release Cause. Use the parameter DefaultCauseMapISDN2IP (described in Section 6.5.9 on page 237) to define a default ISDN Cause that is always used except when the following Release Causes are received: Normal Call Clearing (16), User Busy (17), No User Responding (18) or No Answer from User (19). This mechanism is only available for Tel to IP calls.

H.1 Reason Header

The gateway supports the Reason header according to RFC 3326. The Reason header is used to convey information describing the disconnection cause of a call:

- Sending Reason header: If a call is disconnected from the Tel side (ISDN), the Reason header is set to the received Q.850 cause in the appropriate message (BYE / CANCEL / final failure response) and sent to the SIP side. If the call is disconnected because of a SIP reason, the Reason header is set to the appropriate SIP response.
- Receiving Reason header: If a call is disconnected from the IP side and the SIP message includes the Reason header, it is sent to the Tel side according to the following logic:
    - If the Reason header includes a Q.850 cause, it is sent as is.
    - If the Reason header includes a SIP response: (1) If the message is a final response, the response status code is translated to Q.850 format and passed to ISDN. (2) If the message isn’t a final response, it is translated to a Q.850 cause.
    - When the Reason header is received twice (i.e., SIP Reason and Q.850), the Q.850 takes precedence over the SIP reason and is sent to the Tel side.

H.2 Fixed Mapping of ISDN Release Reason to SIP Response

Table H-1 below describes the mapping of ISDN release reason to SIP response.

Table H-1: Mapping of ISDN Release Reason to SIP Response (continues on pages 446 to 447)

ISDN Release Reason	Description	SIP Response	Description
1	Unallocated number	404	Not found
2	No route to network	404	Not found
3	No route to destination	404	Not found
6	Channel unacceptable	406*	Not acceptable
7	Call awarded and being delivered in an established channel	500	Server internal error
16	Normal call clearing	-*	BYE
17	User busy	486	Busy here
18	No user responding	408	Request timeout
19	No answer from the user	480	Temporarily unavailable
20	Subscriber absent	480	Temporarily unavailable
21	Call rejected	403	Forbidden
22	Number changed w/o diagnostic	410	Gone
22	Number changed with diagnostic	410	Gone
23	Redirection to new destination	480	Temporarily unavailable
26	Non-selected user clearing	404	Not found
27	Destination out of order	502	Bad gateway
28	Address incomplete	484	Address incomplete
29	Facility rejected	501	Not implemented
30	Response to status enquiry	501*	Not implemented
31	Normal unspecified	480	Temporarily unavailable
34	No circuit available	503	Service unavailable
38	Network out of order	503	Service unavailable
41	Temporary failure	503	Service unavailable
42	Switching equipment congestion	503	Service unavailable
43	Access information discarded	502*	Bad gateway
44	Requested channel not available	503*	Service unavailable
47	Resource unavailable	503	Service unavailable
49	QoS unavailable	503*	Service unavailable

* Messages and responses were created since the 'ISUP to SIP Mapping' draft doesn't specify their cause code mapping.

Table H-1: Mapping of ISDN Release Reason to SIP Response (continues on pages 446 to 447)

ISDN Release Reason	Description	SIP Response	Description
50	Facility not subscribed	503*	Service unavailable
55	Incoming calls barred within CUG	403	Forbidden
57	Bearer capability not authorized	403	Forbidden
58	Bearer capability not presently available	503	Service unavailable
63	Service/option not available	503*	Service unavailable
65	Bearer capability not implemented	501	Not implemented
66	Channel type not implemented	480*	Temporarily unavailable
69	Requested facility not implemented	503*	Service unavailable
70	Only restricted digital information bearer capability is available	503*	Service unavailable
79	Service or option not implemented	501	Not implemented
81	Invalid call reference value	502*	Bad gateway
82	Identified channel does not exist	502*	Bad gateway
83	Suspended call exists, but this call identity does not	503*	Service unavailable
84	Call identity in use	503*	Service unavailable
85	No call suspended	503*	Service unavailable
86	Call having the requested call identity has been cleared	408*	Request timeout
87	User not member of CUG	503	Service unavailable
88	Incompatible destination	503	Service unavailable
91	Invalid transit network selection	502*	Bad gateway
95	Invalid message	503	Service unavailable
96	Mandatory information element is missing	409*	Conflict
97	Message type non-existent or not implemented	480*	Temporarily not available
98	Message not compatible with call state or message type non-existent or not implemented	409*	Conflict
99	Information element non-existent or not implemented	480*	Not found
100	Invalid information elements contents	501*	Not implemented
101	Message not compatible with call state	503*	Service unavailable
102	Recovery of timer expiry	408	Request timeout
111	Protocol error	500	Server internal error
127	Interworking unspecified	500	Server internal error

H.3 Fixed Mapping of SIP Response to ISDN Release Reason

Table H-2 below describes the mapping of SIP response to ISDN release reason.

Table H-2: Mapping of SIP Response to ISDN Release Reason (continues on pages 448 to 449)

SIP Response	Description	ISDN Release Reason	Description
400*	Bad request	31	Normal, unspecified
401	Unauthorized	21	Call rejected
402	Payment required	21	Call rejected
403	Forbidden	21	Call rejected
404	Not found	1	Unallocated number
405	Method not allowed	63	Service/option unavailable
406	Not acceptable	79	Service/option not implemented
407	Proxy authentication required	21	Call rejected
408	Request timeout	102	Recovery on timer expiry
409	Conflict	41	Temporary failure
410	Gone	22	Number changed w/o diagnostic
411	Length required	127	Interworking
413	Request entity too long	127	Interworking
414	Request URI too long	127	Interworking
415	Unsupported media type	79	Service/option not implemented
420	Bad extension	127	Interworking
480	Temporarily unavailable	18	No user responding
481*	Call leg/transaction doesn’t exist	127	Interworking
482*	Loop detected	127	Interworking
483	Too many hops	25	Exchange – routing error
484	Address incomplete	28	Invalid number format
485	Ambiguous	1	Unallocated number
486	Busy here	17	User busy
488	Not acceptable here	31	Normal, unspecified
500	Server internal error	41	Temporary failure
501	Not implemented	38	Network out of order
502	Bad gateway	38	Network out of order
503	Service unavailable	41	Temporary failure

* Messages and responses were created since the ‘ISUP to SIP Mapping’ draft doesn’t specify their cause code mapping.

Table H-2: Mapping of SIP Response to ISDN Release Reason (continues on pages 448 to 449)

SIP Response	Description	ISDN Release Reason	Description
504	Server timeout	102	Recovery on timer expiry
505*	Version not supported	127	Interworking
600	Busy everywhere	17	User busy
603	Decline	21	Call rejected
604	Does not exist anywhere	1	Unallocated number
606*	Not acceptable	38	Network out of order

Reader's Notes

I SNMP Traps

This section provides information on proprietary SNMP traps currently supported by the gateway. There is a separation between traps that are alarms and traps that are not (logs). Currently all have the same structure made up of the same 11 varbinds (Variable Binding) (1.3.6.1.4.1.5003.9.10.1.21.1).

The source varbind is composed of a string that details the component from which the trap is being sent (forwarded by the hierarchy in which it resides). For example, an alarm from an SS7 link has the following string in its source varbind:

```
acBoard#1/SS7#0/SS7Link#6
```

In this example, the SS7 link number is specified as 6 and is part of the only SS7 module in the device that is placed in slot number 1 (in a chassis) and is the module to which this trap relates. For devices where there are no chassis options the slot number of the gateway is always 1.

I.1 Alarm Traps

The following tables provide information on alarms that are raised as a result of a generated SNMP trap. The component name (described in each of the following headings) refers to the string that is provided in the 'acBoardTrapGlobalsSource' trap varbind. To clear a generated alarm the same notification type is sent but with the severity set to 'cleared'.

I.1.1 Component: Board#<n>

The source varbind text for all the alarms under this component is Board#<n> where n is the slot number (i.e., <n> = 1).

Table I-1: acBoardFatalError Alarm Trap

Alarm:	acBoardFatalError
OID:	1.3.6.1.4.1.5003.9.10.1.21.2.0.1
Default Severity:	Critical
Event Type:	equipmentAlarm
Probable Cause:	underlyingResourceUnavailable (56)
Alarm Text:	Board Fatal Error: <text>
Status Changes:	
Condition:	Any fatal error
Alarm status:	Critical
<text> value:	A run-time specific string describing the fatal error
Condition:	After fatal error
Alarm status:	Status stays critical until reboot. A clear trap is not sent.
Corrective Action:	Capture the alarm information and the Syslog clause, if active. Contact your first-level support group. The support group will likely want to collect additional data from the device and perform a reset.

Table I-2: acBoardConfigurationError Alarm Trap

Alarm:	acBoardConfigurationError
OID:	1.3.6.1.4.1.5003.9.10.1.21.2.0.2
Default Severity	Critical
Event Type:	equipmentAlarm
Probable Cause:	underlyingResourceUnavailable (56)
Alarm Text:	Board Config Error: <text>
Status Changes:	
Condition:	A configuration error was detected
Alarm status:	critical
<text> value:	A run-time specific string describing the configuration error.
Condition:	After configuration error
Alarm status:	Status stays critical until reboot. A clear trap is not sent.
Corrective Action:	Inspect the run-time specific string to determine the nature of the configuration error. Fix the configuration error using the appropriate tool: Web interface, EMS, or *ini* file. Save the configuration and if necessary reset the device.

Table I-3: acBoardTemperatureAlarm Alarm Trap

Alarm:	acBoardTemperatureAlarm (doesn't apply to the TP-260 board)
OID:	1.3.6.1.4.1.5003.9.10.1.21.2.0.3
Default Severity	Critical
Event Type:	equipmentAlarm
Probable Cause:	temperatureUnacceptable (50)
Alarm Text:	Board temperature too high
Status Changes:	
Condition:	Temperature is above 60 degrees C (140 degrees F)
Alarm status:	Critical
Condition:	After raise, temperature falls below 55 degrees C (131 degrees F)
Alarm status:	Cleared
Corrective Action:	Inspect the system. Determine if all fans in the system are properly operating.

Table I-4: acBoardEvResettingBoard Alarm Trap

Alarm:	acBoardEvResettingBoard
OID:	1.3.6.1.4.1.5003.9.10.1.21.2.0.5
Default Severity:	critical
Event Type:	equipmentAlarm
Probable Cause:	outOfService (71)
Alarm Text:	User resetting board
Status Changes:	
Condition:	When a soft reset is triggered via the Web interface or SNMP.
Alarm status:	Critical
Condition:	After raise
Alarm status:	Status stays critical until reboot. A clear trap is not sent.
Corrective Action:	A network administrator has taken action to reset the device. No corrective action is required.

Table I-5: acFeatureKeyError Alarm Trap

Alarm:	acFeatureKeyError
OID:	1.3.6.1.4.1.5003.9.10.1.21.2.0.6
Default Severity	Critical
Event Type:	processingErrorAlarm
Probable Cause:	configurationOrCustomizationError (7)
Alarm Text:	Feature key error
Note:	Support for this alarm is pending

Table I-6: acBoardCallResourcesAlarm Alarm Trap

Alarm:	acBoardCallResourcesAlarm
OID:	1.3.6.1.4.1.5003.9.10.1.21.2.0.8
Default Severity:	Major
Event Type:	processingErrorAlarm
Probable Cause:	softwareError (46)
Alarm Text:	Call resources alarm
Status Changes:	
Condition:	Number of free channels exceeds the predefined RAI *high* threshold.
Alarm Status:	Major
Note:	To enable this alarm the RAI mechanism must be activated (EnableRAI = 1).
Condition:	Number of free channels falls below the predefined RAI *low* threshold.
Alarm Status:	Cleared

Table I-7: acBoardControllerFailureAlarm Alarm Trap

Alarm:	acBoardControllerFailureAlarm
OID:	1.3.6.1.4.1.5003.9.10.1.21.2.0.9
Default Severity:	Minor
Event Type:	processingErrorAlarm
Probable Cause:	softwareError (46)
Alarm Text:	Controller failure alarm
Status Changes:	
Condition:	Proxy has not been found
Alarm Status:	Major
Additional Info:	Proxy not found. Use internal routing, or Proxy lost. Looking for another Proxy
Condition:	Proxy is found. The clear message includes the IP address of the located Proxy.
Alarm Status:	Cleared

Table I-8: acBoardOverloadAlarm Alarm Trap

Alarm:	acBoardOverloadAlarm
OID:	1.3.6.1.4.1.5003.9.10.1.21.2.0.11
Default Severity:	Major
Event Type:	processingErrorAlarm
Probable Cause:	softwareError (46)
Alarm Text:	Board overload alarm
Status Changes:	
Condition:	An overload condition exists in one or more of the system components.
Alarm Status:	Major
Condition:	The overload condition passed
Alarm Status:	Cleared

Table I-9: acgwAdminStateChange Alarm Trap

Alarm:	acgwAdminStateChange
OID:	1.3.6.1.4.1.5003.9.10.1.21.2.0.7
Default Severity	Major
Event Type:	processingErrorAlarm
Probable Cause:	outOfService (71)
Alarm Text:	Network element admin state change alarm Gateway is <text>
Status Changes:	
Condition:	Admin state changed to shutting down
Alarm status:	Major
<text value>:	Shutting down. No time limit.
Condition:	Admin state changed to locked
Alarm status:	Major
<text value>:	Locked
Condition:	Admin state changed to unlocked
Alarm status:	Cleared
Corrective Action:	A network administrator has taken an action to lock the device. No corrective action is required.

Table I-10: acOperationalStateChange Alarm Trap

Alarm:	acOperationalStateChange
OID:	1.3.6.1.4.1.5003.9.10.1.21.2.0.15
Default Severity	Major
Event Type:	processingErrorAlarm
Probable Cause:	outOfService (71)
Alarm Text:	Network element operational state change alarm. Operational state is disabled.
Note:	This alarm is raised if the operational state of the node goes to disabled. The alarm is cleared when the operational state of the node goes to enabled. In IP systems, the operational state of the node is disabled if the device fails to properly initialize.
Status Changes:	
Condition:	Operational state changed to disabled
Alarm status:	Major
Condition:	Operational state changed to enabled
Alarm status:	Cleared
Corrective Action:	In IP systems, check for initialization errors. Look for other alarms and Syslogs that might provide additional information about the error.

I.1.2 Component: AlarmManager#0

The source varbind text for all the alarms under this component is *Board#*<n>*/AlarmManager#0* where n is the slot number.

Table I-11: acActiveAlarmTableOverflow Alarm Trap

Alarm:	acActiveAlarmTableOverflow
OID:	1.3.6.1.4.15003.9.10.1.21.2.0.12
Default Severity:	Major
Event Type:	processingErrorAlarm
Probable Cause:	resourceAtOrNearingCapacity (43)
Alarm Text:	Active alarm table overflow
Status Changes:	
Condition:	Too many alarms to fit in the active alarm table
Alarm status:	Major
Condition:	After raise
Alarm status:	Status stays major until reboot. A clear trap is not sent.
Note:	The status stays major until reboot as it denotes a possible loss of information until the next reboot. If an alarm is raised when the table is full, it is possible that the alarm is active, but does not appear in the active alarm table.
Corrective Action:	Some alarm information may have been lost, but the ability of the device to perform its basic operations has not been impacted. A reboot is the only way to completely clear a problem with the active alarm table. Contact your first-level group.

I.1.3 Component: Chassis#0

The source varbind text for the alarm under the component below is Chassis#0/FanTray#0.

Table I-12: acFanTrayAlarm Alarm Trap

Alarm:	acFanTrayAlarm
OID:	1.3.6.1.4.1.5003.9.10.1.21.2.0.29
Default Severity:	Critical
Event Type:	equipmentAlarm
Probable Cause:	heatingVentCoolingSystemProblem
Alarm Text:	Fan-Tray Alarm.
Status Changes:	
Condition:	Fan-Tray is missing
Alarm status:	Critical
<text value>:	Fan-Tray Alarm. Fan-Tray is missing.
Condition:	One or more fans in the Fan-Tray are faulty.
Alarm status:	Major
<text value>:	Fan tray is in place and fans are working.
Condition:	Fan is faulty.
Alarm status:	Cleared

The source varbind text for the alarm under this component is Chassis#0/PowerSupply#<m>, where *m* is the power supply's slot number.

Table I-13: acPowerSupplyAlarm Alarm Trap

Alarm:	acPowerSupplyAlarm
OID:	1.3.6.1.4.1.5003.9.10.1.21.2.0.30
Default Severity:	Critical
Event Type:	equipmentAlarm
Probable Cause:	powerProblem
Alarm Text:	Power-Supply Alarm. Power-Supply is missing.
Status Changes:	
Condition:	The HA (High Availability) feature is active and one of the power supply units is faulty or missing.
Alarm status:	Major
Condition:	PS unit is placed and working.
Alarm status:	Cleared

I.1.4 Component: EthernetLink#0

The source varbind text for all the alarms under this component is *Board#*<n>*/EthernetLink#0* where n is the slot number. This trap relates to the Ethernet Link Module (the #0 numbering doesn't apply to the physical Ethernet link).

Table I-14: acBoardEthernetLinkAlarm Alarm Trap

Alarm:	acBoardEthernetLinkAlarm
OID:	1.3.6.1.4.1.5003.9.10.1.21.2.0.10
Default Severity:	Critical
Event Type:	equipmentAlarm
Probable Cause:	underlyingResourceUnavailable (56)
Alarm Text:	Ethernet link alarm: <text>
Status Changes:	
Condition:	Fault on single interface
Alarm status:	major
<text> value:	Redundant link is down
Condition:	Fault on both interfaces
Alarm status:	critical
<text> value:	No Ethernet link
Condition:	Both interfaces are operational
Alarm status:	cleared
Corrective Action:	Ensure that both Ethernet cables are plugged into the back of the system. Inspect the system's Ethernet link lights to determine which interface is failing. Reconnect the cable or fix the network problem

I.1.5 Log Traps (Notifications)

This section details traps that are not alarms. These traps are sent with the severity varbind value of ‘indeterminate’. These traps don’t ‘clear’, they don’t appear in the alarm history or active tables. One log trap that does send clear is acPerformanceMonitoringThresholdCrossing.

Table I-15: acKeepAlive Log Trap

Trap:	acKeepAlive
OID:	1.3.6.1.4.1.5003.9.10.1.21.2.0.16
Default Severity:	Indeterminate
Event Type:	other (0)
Probable Cause:	other (0)
Trap Text:	Keep alive trap
Status Changes:	
Condition:	The STUN client in is enabled and identified a NAT device or doesn’t locate the STUN server. The *ini* file contains the following line: ‘SendKeepAliveTrap=1’
Trap Status:	Trap is sent
Note:	Keep-alive is sent every 9/10 of the time defined in the parameter NatBindingDefaultTimeout.

Table I-16: acPerformanceMonitoringThresholdCrossing Log Trap

Trap:	acPerformanceMonitoringThresholdCrossing
OID:	1.3.6.1.4.1.5003.9.10.1.21.2.0.27
Default Severity:	Indeterminate
Event Type:	other (0)
Probable Cause:	other (0)
Trap Text:	"Performance: Threshold trap was set”, with source = name of performance counter which caused the trap
Status Changes:	
Condition:	A performance counter has crossed the high threshold
Trap Status:	Indeterminate
Condition:	A performance counter has crossed the low threshold
Trap status:	cleared

Table I-17: acHTTPDownloadResult Log Trap

Trap:	acHTTPDownloadResult
OID:	1.3.6.1.4.1.5003.9.10.1.21.2.0.28
Default Severity:	Indeterminate
Event Type:	processingErrorAlarm (3) for failures and other (0) for success.
Probable Cause:	other (0)
Status Changes:	
Condition:	Successful HTTP download.
Trap Text:	HTTP Download successful
Condition:	Failed download.
Trap Text:	HTTP download failed, a network error occurred.
Note:	There are other possible textual messages describing NFS failures or success, FTP failure or success.

I.1.6 Other Traps

The following are provided as SNMP traps and are not alarms.

Table I-18: coldStart Trap

Trap Name:	coldStart
OID:	1.3.6.1.6.3.1.1.5.1
MIB:	SNMPv2-MIB
Note:	This is a trap from the standard SNMP MIB.

Table I-19: authenticationFailure Trap

Trap Name:	authenticationFailure
OID:	1.3.6.1.6.3.1.1.5.5
MIB:	SNMPv2-MIB

Table I-20: acBoardEvBoardStarted Trap

Trap Name:	acBoardEvBoardStarted
OID:	1.3.6.1.4.1.5003.9.10.1.21.2.0.4
MIB:	AcBoard
Severity:	cleared
Event Type:	equipmentAlarm
Probable Cause:	Other(0)
Alarm Text:	Initialization Ended
Note:	This is the AudioCodes Enterprise application cold start trap.

Table I-21: AcDChannelStatus Trap

Trap Name:	acDChannelStatus
OID:	1.3.6.1.4.1.5003.9.10.1.21.2.0.37
MIB	AcBoard
Severity	Minor
Event Type:	communicationsAlarm
Probable Cause:	communicationsProtocolError
Alarm Text:	D-Channel Trap.
Source:	Trunk <m> where m is the trunk number (starts from 0).
Status Changes:	
Condition:	D-Channel un-established.
Trap status:	Trap is sent with the severity of Minor.
Condition:	D-Channel established.
Trap status:	Trap is sent with the severity of Cleared.

I.1.7 Trap Varbinds

Each trap described above provides the following fields (known as ‘varbinds’). Refer to the AcBoard MIB for additional details on these varbinds.

- acBoardTrapGlobalsName
- acBoardTrapGlobalsTextualDescription
- acBoardTrapGlobalsSource
- acBoardTrapGlobalsSeverity
- acBoardTrapGlobalsUniqID
- acBoardTrapGlobalsType
- acBoardTrapGlobalsDateAndTime
- acBoardTrapGlobalsProbableCause
- acBoardTrapGlobalsAdditionalInfo1
- acBoardTrapGlobalsAdditionalInfo2
- acBoardTrapGlobalsAdditionalInfo3

Note that ‘acBoardTrapGlobalsName’ is actually a number. The value of this varbind is ‘X’ minus 1, where ‘X’ is the last number in the trap’s OID. For example, the ‘name’ of ‘acBoardEthernetLinkAlarm’ is ‘9’. The OID for ‘acBoardEthernetLinkAlarm’ is 1.3.6.1.4.1.5003. 9.10.1.21.2.0.10.

J Installation and Configuration of Apache HTTP Server

This appendix describes the installation and configuration of Apache's HTTP server with Perl script environment (required for recording).

J.1 Windows 2000/XP Operation Systems

Note: For detailed installation information, refer to http://perl.apache.org/docs/2.0/os/win32/config.html.

Additional software (supplied with the software package) is required: an uploading script (put.cgi), supplied.

➢ To configure the Apache HTTP server and mod_perl version 2.0 software, take these 9 steps:

1. Download the third party Perl-5.8-win32-bin.exe, installation file, from the following link: www.apache.org/dist/perl/win32-bin/Perl-5.8-win32-bin.exe. The installation file includes: Apache 2.0.46, Perl 5.8.0 and mod_perl-1.99 (the content of the file and the software version are subject to modification and changes in the future).

   For full installation instructions refer to www.apache.org/dist/perl/win32-bin/Perl-5.8-win32-bin.readme.

2. To start the installation wizard run the Perl-5.8-win32-bin.exe file.
3. During the installing, you are prompt to determine the Destination Folder under which the package is installed, it is advised to provide a non-spaced path (such as: c:\\directory_name_without_spaces).
4. In the following screen (configuration): uncheck the "Build html docs" and "Configure CPAN pm" checkboxes, if they are present. If you are prompted to bring "nmake" answer no.
5. After the installation is complete, add the "/path/perl/bin" and "/path/apache2/bin" (*path* stands for the path that was previously specified in the "Destination Folder") directories to the system known path. Open the Control Panel→System→Advanced→Environment Variables, inside the System Variables dialog box choose "Path" and click the Edit button; in the opened Variable Value checkbox append both of the paths to the existing list. Restart window in order to activate the new paths.
6. Open the Apache2/conf/httpd.conf file for editing and set the parameter MaxKeepAliveRequests to 0 (enables an unlimited number of requests during a persistent connection – required for multiple consecutive HTTP PUT requests for uploading the file).

7. Open the Apache2/conf/perl.conf file for editing and add the line “Script PUT /perl/put.cgi” after the last line in the following section (note that if the following section is omitted or different in the file, insert it into the file or change it there accordingly):

```
Alias /perl/ "C:/Apache2/perl/
<Location /perl>
SetHandler perl-script
PerlResponseHandler ModPerl::Registry
Options +ExecCGI
PerlOptions +ParseHeaders
</Location>
```

8. Locate the file put.cgi on the supplied software package and copy it into the Apach2\perl\ directory. Change the first line in this file from c:/perl/bin/perl to your perl executable file (this step is required only if mod_perl is not included in your Apache installation).
9. In the apach2\bin directory, from a DOS prompt, type the following commands: Apache.exe -n Apache2 -k install
Apache.exe -n Apache2 -k start

The installation and configuration are finished. You are now ready to start using the HTTP server.

J.2 Linux Operation Systems

Note: It is assumed that the installing of Linux already includes: Apache server (for example, Apache 1.3.23), perl and mod_perl (for example mod_perl 1.26).

Additional required software: an uploading script (put.cgi), supplied with the software package.

➢ To configure Apache HTTP server, take these 4 steps:

1. Inside the Apache directory, create the directory /perl (for example /var/www/perl). Locate the file put.cgi on the supplied software package and copy it to that directory.
2. In the put.cgi script, change the first line from c:/perl/bin/perl to your perl executable file (this step is required only if mod_perl is not included in your Apache installation).
3. Enable access to the following directories and files by typing:
   - >chmod 777 perl
   - >chmod 755 put.cgi
   - >chmod 777 html (the name of the server’s shared files directory)
4. Configure the Apache sever:
   a. Open etc/httpd/conf/httpd.conf (or a similar file) for editing
   b. Set the KeepAlive parameter to true
   c. Set the MaxKeepAliveRequests parameter to 0 (enables an unlimited number of requests during a persistent connection – required for multiple consecutive HTTP POST requests for uploading the file).

d. Set MaxClients to 250

e. Change the mod_perl module lines to:

```
<IfModule mod_perl.c>
    Alias /perl/ /var/www/perl/
    <Directory /var/www/perl>
        SetHandler perl-script
        PerlHandler Apache::Registry
        Options +ExecCGI
        PerlSendHeader On
    </Directory>
</IfModule>

Script PUT /perl/put.cgi
```

Reader's Notes

K OSN Server Hardware Installation

This section is intended for customers who have already purchased the Mediant 1000 Media Gateway and want to install the added Mediant 1000 OSN (Open Solution Network) Server functionality. The OSN server module hosts third-party VoIP applications such as IP-PBX, Pre-Paid, and IP-PBX redundancy.

The OSN server is a standalone entity, integrated within the Mediant 1000 gateway, using a separate Ethernet interface and IP configuration as that used for the gateway.

K.1 Required Working Tools

The following tools are required for installing the OSN Server module:

- Phillips screwdriver
- Flathead screwdriver
- Wire cutter

K.2 OSN Server Installation on the Mediant 1000

The Mediant 1000 OSN Server package is composed of three modules, which need to be installed in the Mediant 1000:

- Connection Module (CM)
- iPMX
- Hard Drive Module (HDMX)

Warning: Before installing the Mediant 1000 OSN Server modules, ensure that the Mediant 1000 is disconnected from the power supply. These modules are not hot-swappable and damage to these modules can occur if replaced under voltage.

These modules are shown in the figures below.

Figure K-1: Connection (CM) Module

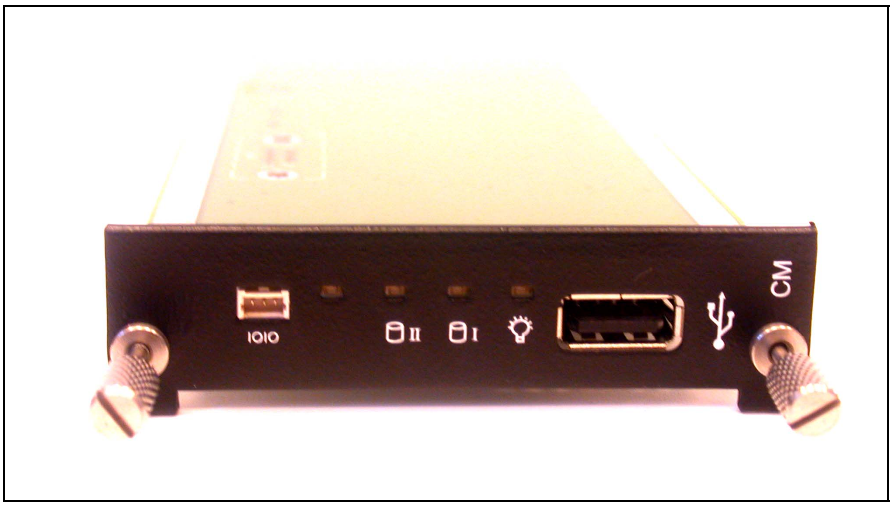


Figure K-2: iPMX Module

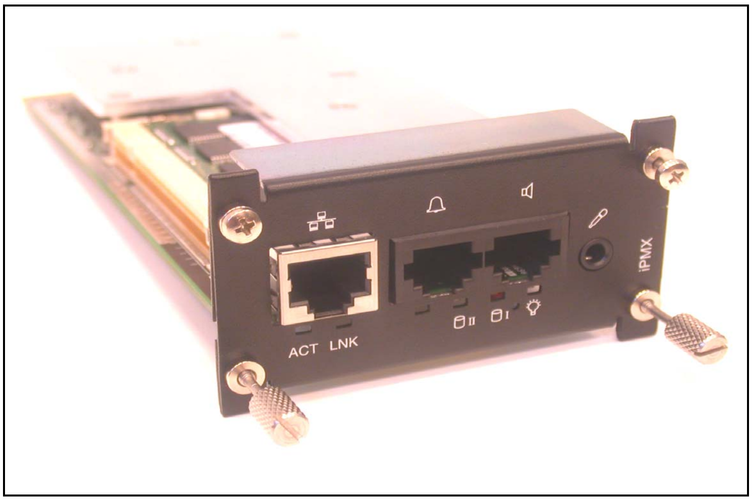
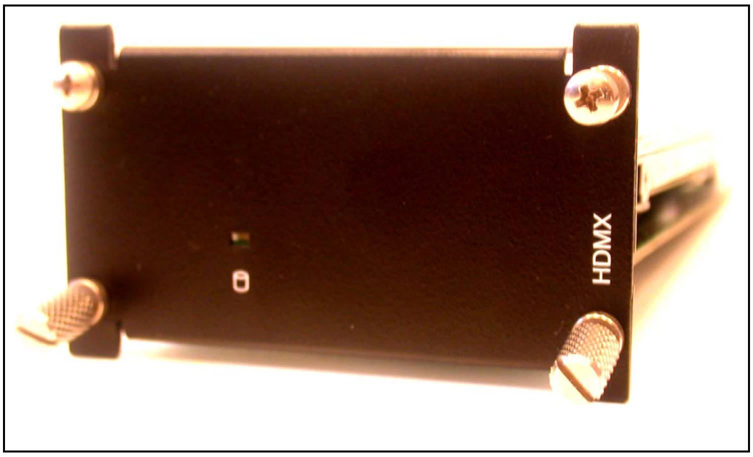


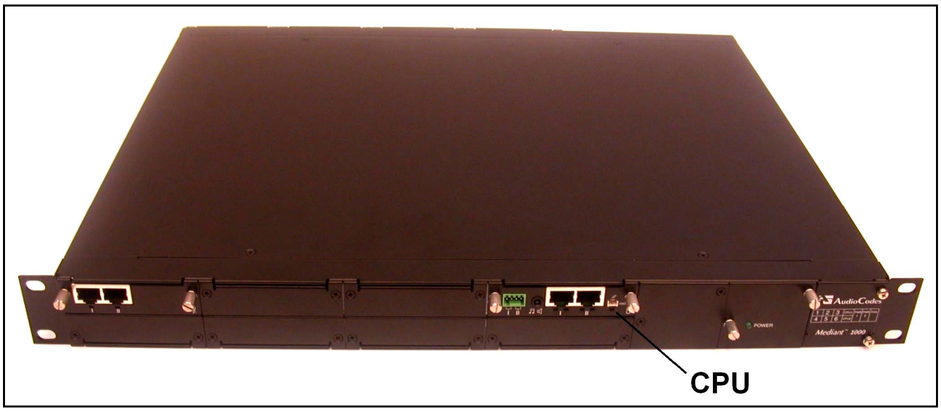
Figure K-3: Hard Drive (HDMX) Module



➢ To install the CM module, take these 5 steps:

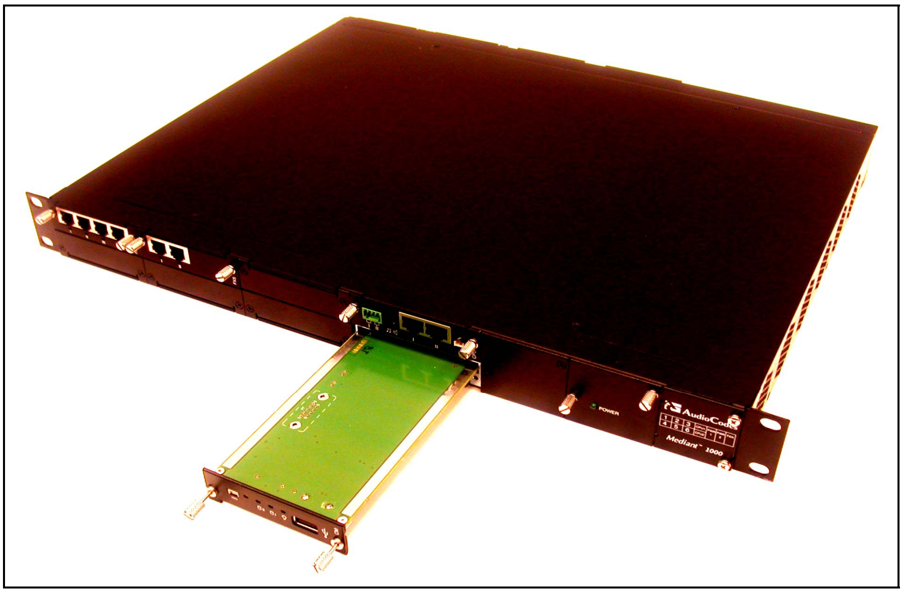
1. Place the Mediant 1000, so that the front panel is facing you, as shown in Figure K-4.
2. Using a Phillips screwdriver, remove the black metal cover plate from the slot below the module labeled CPU, as shown in Figure K-4.

Figure K-4: Mediant 1000 Front Panel View



3. Insert the CM module in the slot below the CPU, with the plain side of the Printed Circuit Board (PCB) facing up. Make sure the PCB slides into the slot rails. (As the slot rails are not easily visible, it is important to align the CM with the rails in the slot, before sliding it in).

Figure K-5: Inserting the CM Module

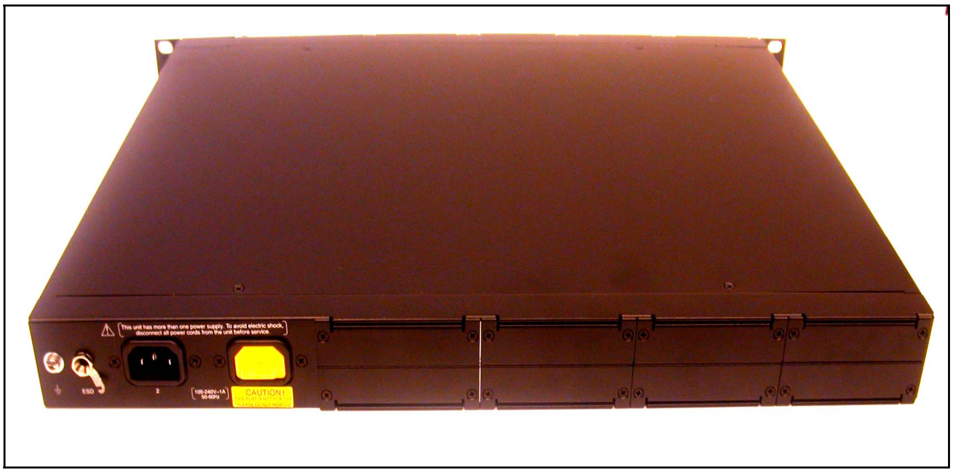


4. Push the CM module into the slot and press on it firmly to ensure it has been fully inserted.
5. Using a straight regular screwdriver, tighten the mounting pins.

➢ To install the iPMX module, take these 6 steps:

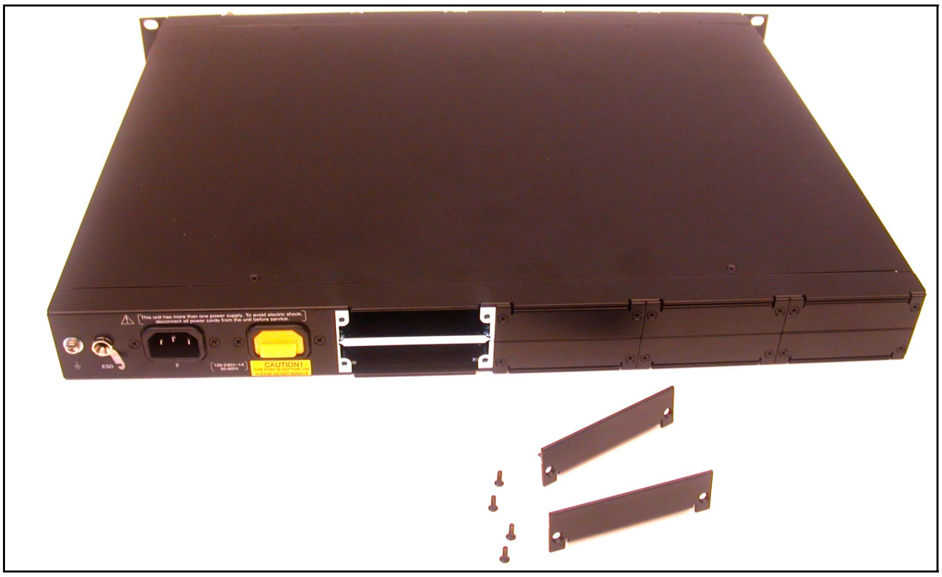
1. Place the Mediant 1000, so that the rear panel is facing you, as shown in Figure K-6.

Figure K-6: Mediant 1000 Rear Panel View



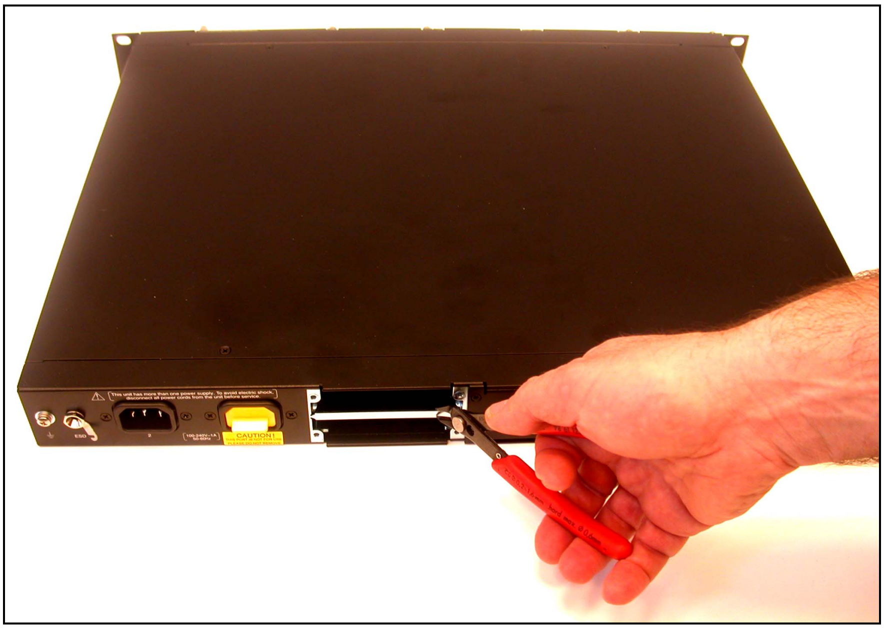
2. Remove the black metal cover plates in the first and second slots located on the right side of the power connection, as shown in Figure K-7.

Figure K-7: Mediant 1000 With Cover Plates Removed



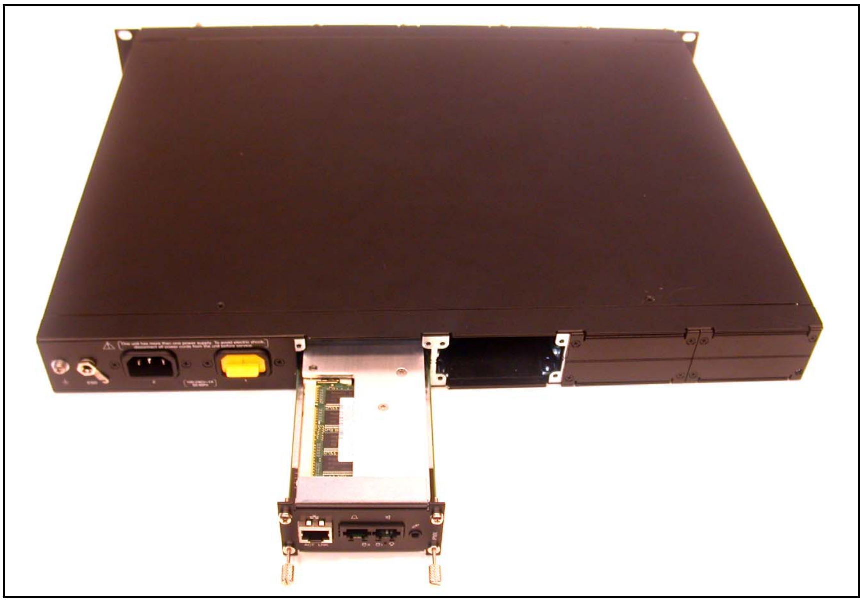
3. Use the cutter tool to remove the small metal strip between the upper and lower slots, as shown in Figure K-8.

Figure K-8: Mediant 1000 With Cutter Tool



4. Insert the iPMX module in the first slot, closest to the power connection, as shown in Figure K-9.

Figure K-9: Inserting the iPMX Module

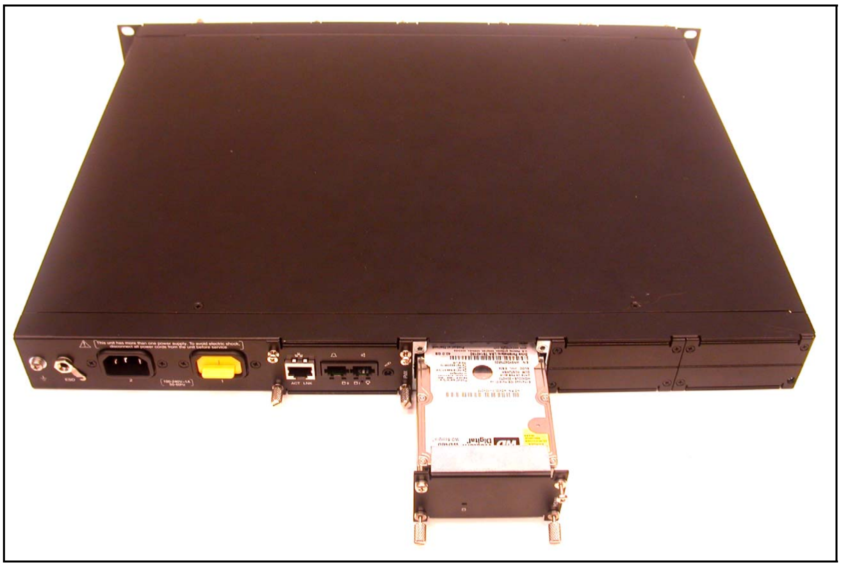


5. Push the iPMX module into the slot and press on it firmly to ensure it has been fully inserted.
6. Using a straight regular screwdriver, tighten the mounting pins.

➢ To install the Hard Drive (HDMX) module, take these 6 steps:

1. Place the Mediant 1000, so that the rear panel is facing you, as shown in Figure K-6.
2. Remove the black metal cover plates in the first and second slots located on the right side of the power connection, as shown in Figure K-7.
3. Use the cutter tool to remove the small metal strip between the upper and lower slots, as shown in Figure K-8.
4. Insert the Hard Drive (HDMX) module in the second slot, as shown in Figure K-10.

Figure K-10: Inserting the HDMX Module



5. Push the Hard Drive (HDMX) module into the slot and press on it firmly to ensure it has been fully inserted.
6. Using a straight regular screwdriver, tighten the mounting pins.

L Linux™ Operating System Installation for the OSN Server

The purpose of this appendix is to describe how to install the various distributions of the Linux^{TM} Operating System, on the Mediant 1000 Media Gateway.

This document targets RedHat^{TM}, Debian^{TM} and SUSE^{TM} distributions. It can also serve as a reference when trying to install other Linux^{TM} distributions. These instructions have been verified against the following distributions:

- RedHat^{TM} Linux 9
- RedHat^{TM} Fedora Core 4
- RedHat^{TM} Enterprise 3
- RedHat^{TM} Enterprise 4
- Debian^{TM} 3.1 (r0a) “Sarge”
- SUSE^{TM} 9.3
- SUSE^{TM} 10

Known issues with the following Linux^{TM} distributions:

- Redhat^{TM} Fedora Core 3 is not supported.

L.1 Physical Mounting Requirements

Before installing LinuxTM on the Mediant 1000 OSN Server, ensure you have the following:

- Ethernet cable cord
- External USB CD ROM or DVD ROM (not supplied by AudioCodes)
- USB cable (not supplied by AudioCodes) to connect the external USB CD ROM to the Mediant 1000
- RS-232 cable (supplied by AudioCodes)
- Linux^{TM} Distributions Installation CDs
- Blank CD or DVD media
- Windows^{TM} PC with CD RW or external CD ROM/RW

L.2 Software Requirements

- RS-232 Console software (HyperTerminal^{TM} or other terminal software)
- ISO Image Editor (WinISO^{TM} is recommended. Refer to http://www.winiso.com/download.htm)
- UNIX^{TM} File Format text editor (Refer to http://www.pspad.com or http://www.ultraedit.com)

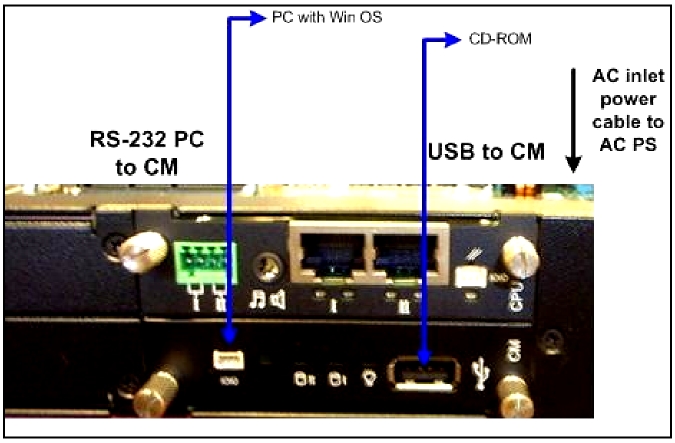
L.3 Physical Mounting Requirements

L.3.1 Mediant 1000 Front Panel Connection

➢ To connect to the front panel, take these 3 steps:

1. Connect the RS-232 cable from the PC to the RS-232 port on the Mediant 1000.
2. Connect the USB connection cable from the external CD-ROM to the USB port. Refer to the figure below.

Figure L-1: Mediant 1000 Front Panel Connections



L.3.2 Mediant 1000 Rear Panel Connection

➢ To connect to the rear panel, take these 3 steps:

1. Connect the Ethernet cable to the ACT Link, on the rear side of the Mediant 1000.
1. Connect the external CD-ROM power supply.
2. Connect the Mediant 1000 power supply.

L.4 Software Installation

The following three sections describe how to install different versions of Linux™.

- Linux™ RedHat (and Fedora) Installation
- Linux™ Debian Installation
- Linux™ SUSE Installation

L.5 Linux™ RedHat (and Fedora) Installation Instructions

Perform the following four stages for Linux™ Redhat (and Fedora) Installation. (Some distributions of Linux™ may vary slightly):

L.5.1 Stage 1: Obtaining the ISO Image

In order to obtain an updated ISO image, either download it from the AudioCodes website as described in Section L.5.1.1, or create it using the steps detailed in Section L.5.1.2.

L.5.1.1 Downloading an updated LinuxTM Redhat ISO Image

➢ **To download an ISO image from the AudioCodes website, take these 6 steps:**

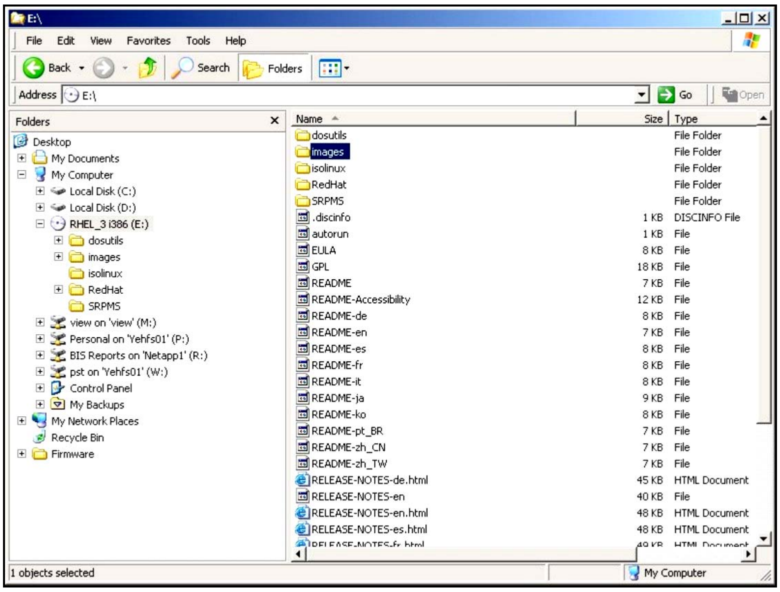
1. From http://www.audiocodes.com, navigate to the Support page.
2. Login with your Username and Password.
3. Under Product Documentation, select the Mediant 1000 link and click on Mediant 1000 OSN Server.
4. Select **Linux Boot Image** and select the installation you need.
5. Download the **ISO image** to a folder called **Partner Install** on your PC.
6. Continue with Stage 3 – Burn the CD (Section L.5.3 on page 479).

L.5.1.2 Creating an updated Linux Redhat ISO Image

➢ **To create an updated Linux Redhat ISO Image take these 6 steps**

1. On the local hard disk of the Window's™ PC, create a new folder called "**Partner Install**".
2. If you have not already done so, download a utility that allows editing of an ISO image. The WinISO™ program (http://www.winiso.com/download.htm) may be used.
3. Using Internet Explorer, download a Unix File Format text editor. PSPad™ (http://www.pspad.com) or UltraEdit™ (http://www.ultraedit.com) may be used.
4. Insert the 1st installation disk of the Linux™ Redhat distribution into the CD-ROM drive of the Windows™ PC. The Windows Explorer screen appears displaying files currently on the CD.

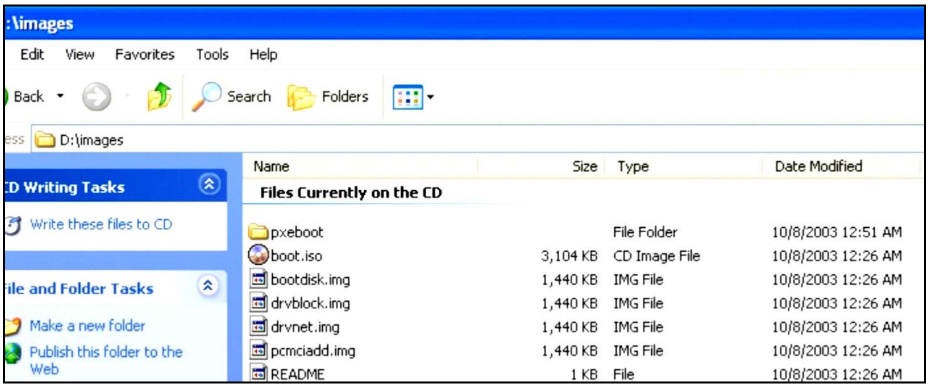
Figure L-2: Disk 1 of Redhat Partner Installation



Name	Size	Type
dosutils		File Folder
images		File Folder
isolinux		File Folder
RedHat		File Folder
SRPMS		File Folder
.discinfo	1 KB	DISCINFO File
autorun	1 KB	File
EULA	8 KB	File
GPL	18 KB	File
README	7 KB	File
README-Accessibility	12 KB	File
README-de	8 KB	File
README-en	7 KB	File
README-es	8 KB	File
README-fr	8 KB	File
README-it	8 KB	File
README-ja	9 KB	File
README-ko	8 KB	File
README-pt_BR	7 KB	File
README-zh_CN	7 KB	File
README-zh_TW	7 KB	File
RELEASE-NOTES-de.html	45 KB	HTML Document
RELEASE-NOTES-en	40 KB	File
RELEASE-NOTES-en.html	48 KB	HTML Document
RELEASE-NOTES-es.html	48 KB	HTML Document
RELEASE-NOTES-fr.html	49 KB	HTML Document

5. Locate the **boot.iso** file in the images folder on the CD.

Figure L-3: "Images" Folder



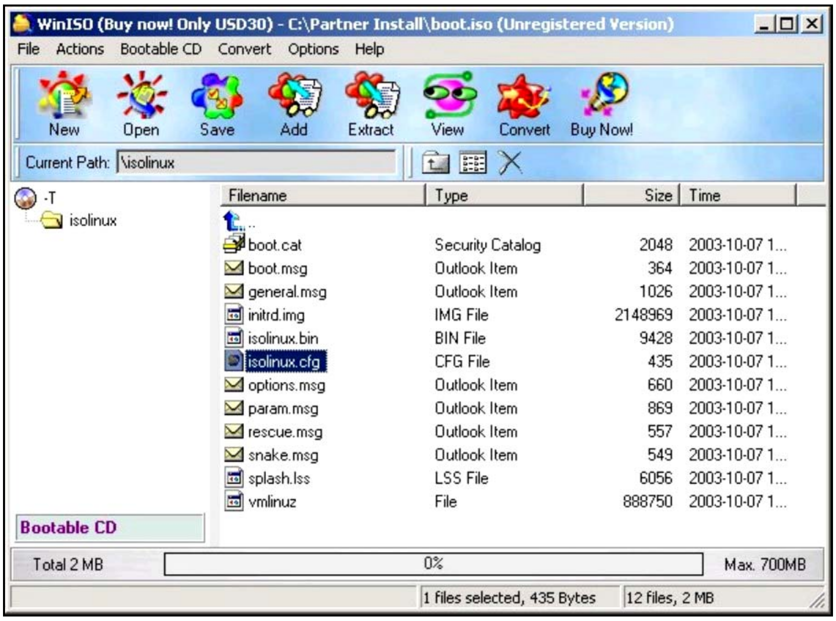
Name	Size	Type	Date Modified
pxeboot		File Folder	10/8/2003 12:51 AM
boot.iso	3,104 KB	CD Image File	10/8/2003 12:26 AM
bootdisk.img	1,440 KB	IMG File	10/8/2003 12:26 AM
drvblock.img	1,440 KB	IMG File	10/8/2003 12:26 AM
drvnet.img	1,440 KB	IMG File	10/8/2003 12:26 AM
pcmciadd.img	1,440 KB	IMG File	10/8/2003 12:26 AM
README	1 KB	File	10/8/2003 12:26 AM

Note: The "images" folder may be named differently on different Linux™ distributions.

6. Copy the **boot.iso** file to the Partner Install folder created in Step 1 and open it with an ISO image editor.

   The **isolinux.cfg** file should appear as shown in the screen below.

Figure L-4: ISO Screen



WinISO (Buy now! Only USD30) - C:\Partner Install\boot.iso (Unregistered Version)
File Actions Bootable CD Convert Options Help
New Open Save Add Extract View Convert Buy Now!
Current Path: \isolinux
-T
isolinux
Bootable CD
Total 2 MB
0%
Max. 700MB
1 files selected, 435 Bytes
12 files, 2 MB

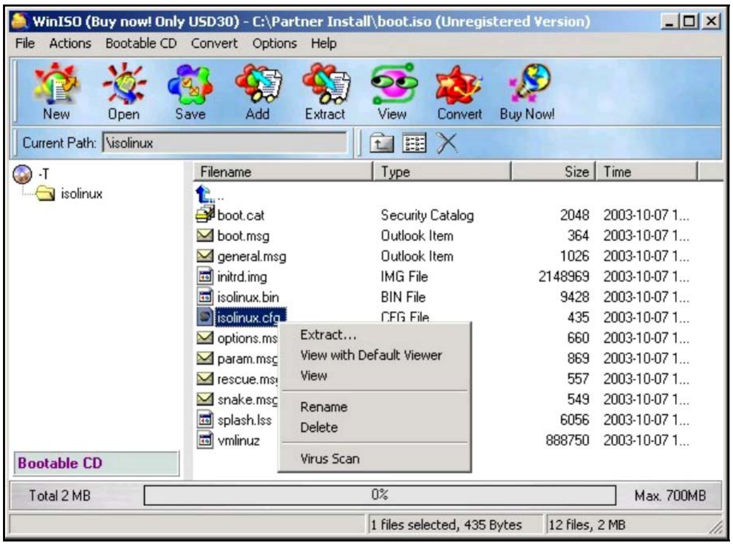
Filename	Type	Size	Time
..			
boot.cat	Security Catalog	2048	2003-10-07 1...
boot.msg	Outlook Item	364	2003-10-07 1...
general.msg	Outlook Item	1026	2003-10-07 1...
initrd.img	IMG File	2148969	2003-10-07 1...
isolinux.bin	BIN File	9428	2003-10-07 1...
isolinux.cfg	CFG File	435	2003-10-07 1...
options.msg	Outlook Item	660	2003-10-07 1...
param.msg	Outlook Item	869	2003-10-07 1...
rescue.msg	Outlook Item	557	2003-10-07 1...
snake.msg	Outlook Item	549	2003-10-07 1...
splash.lss	LSS File	6056	2003-10-07 1...
vmlinuz	File	888750	2003-10-07 1...

L.5.2 Stage 2: Editing the isolinux.cfg File

➢ To edit the isolinux.cfg file, take these 19 steps:

1. Right-click the **isolinux.cfg** file, and then from the shortcut menu, choose **Extract**.

Figure L-5: Selecting Extract Option


WinISO (Buy now! Only USD30) - C:\Partner Install\boot.iso (Unregistered Version)
File Actions Bootable CD Convert Options Help
New Open Save Add Extract View Convert Buy Now!
-T
isolinux
Bootable CD
Extract...
View with Default Viewer
View
Rename
Delete
Virus Scan
Total 2 MB
0%
Max. 700MB

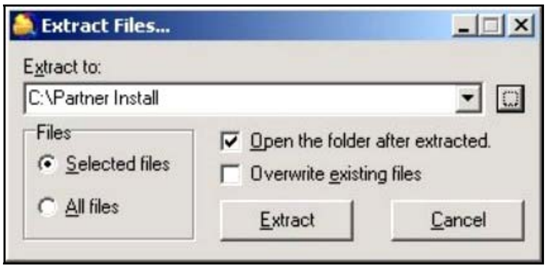
Current Path: \isolinux

Filename	Type	Size	Time
..			
boot.cat	Security Catalog	2048	2003-10-07 1...
boot.msg	Outlook Item	364	2003-10-07 1...
general.msg	Outlook Item	1026	2003-10-07 1...
initrd.img	IMG File	2148969	2003-10-07 1...
isolinux.bin	BIN File	9428	2003-10-07 1...
isolinux.cfg	CFG File	435	2003-10-07 1...
options.ms		660	2003-10-07 1...
param.msg		869	2003-10-07 1...
rescue.ms		557	2003-10-07 1...
snake.msg		549	2003-10-07 1...
splash.lss		6056	2003-10-07 1...
vmlinuz		888750	2003-10-07 1...

1 files selected, 435 Bytes | 12 files, 2 MB

2. Extract the **isolinux.cfg** file to the Partner Install folder created in Stage 1.

Figure L-6: Extract Files



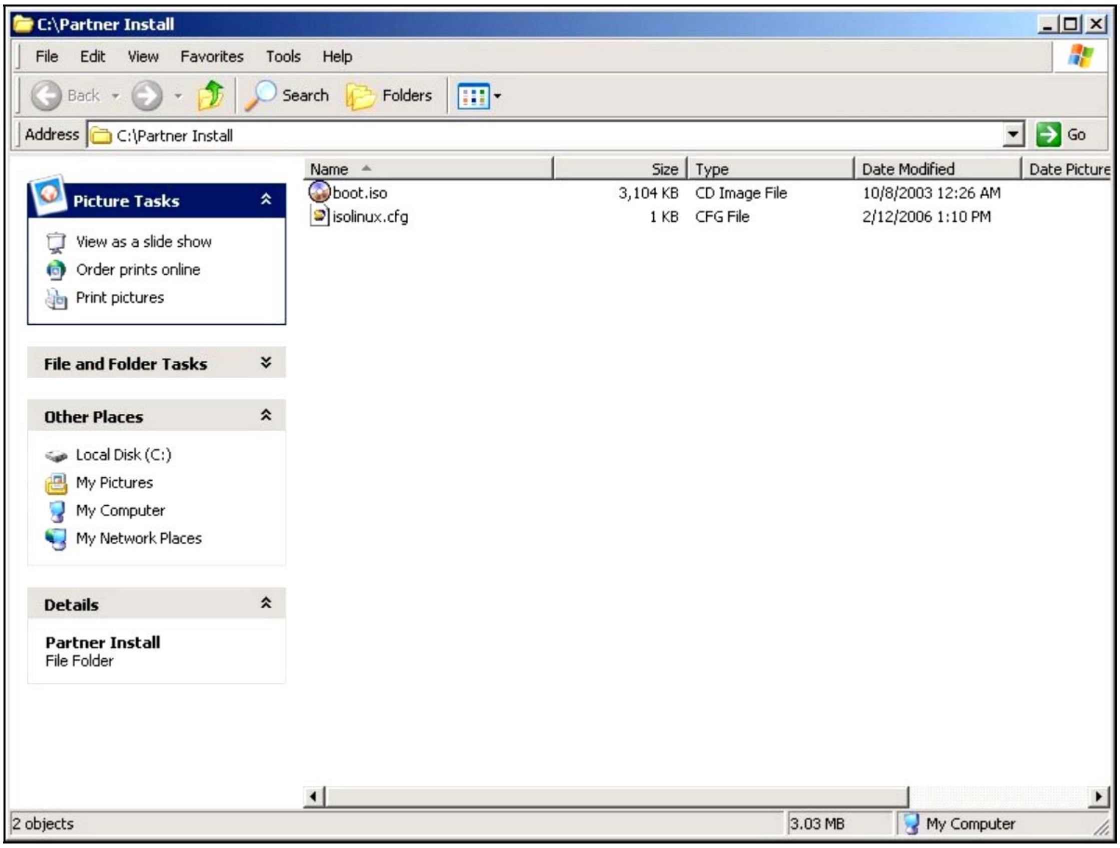
Extract to: C:\Partner Install

Files
● Selected files
○ All files

☑ Open the folder after extracted.
☐ Overwrite existing files

3. Click **Extract**; a screen opens containing the extracted **isolinux** folder.

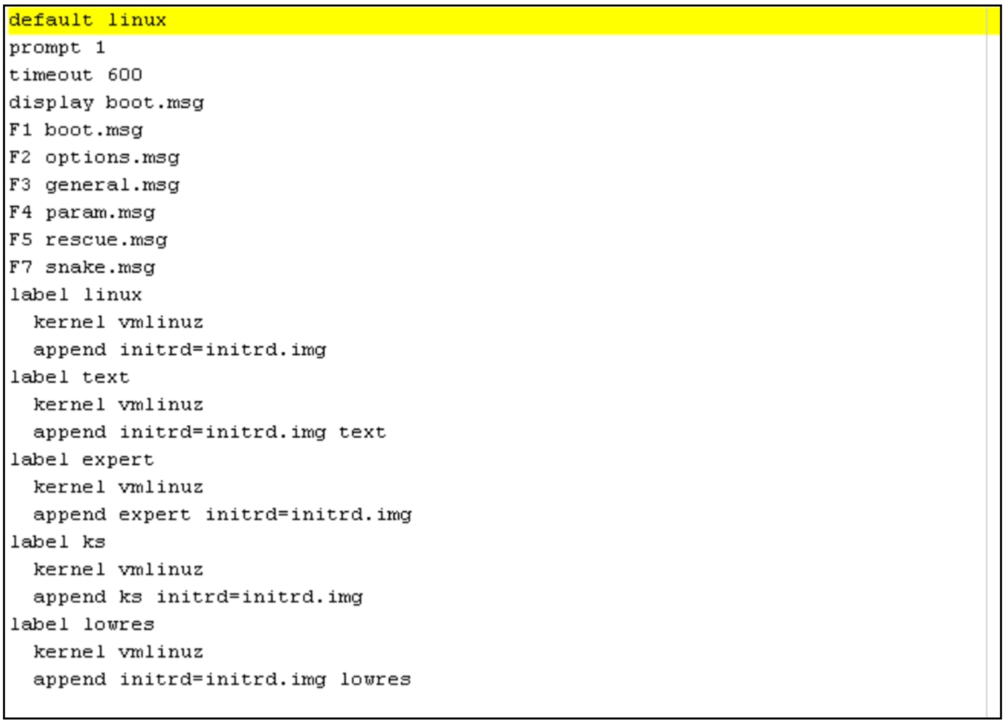
Figure L-7: ISO-Extract screen



Name	Size	Type	Date Modified	Date Picture
boot.iso	3,104 KB	CD Image File	10/8/2003 12:26 AM	
isolinux.cfg	1 KB	CFG File	2/12/2006 1:10 PM	

4. Open the **isolinux.cfg** file with a text editor that supports UNIX file format (e.g., PSPad, UltraEdit). The following screen appears.

Figure L-8: Text Edit screen



```
default linux
prompt 1
timeout 600
display boot.msg
F1 boot.msg
F2 options.msg
F3 general.msg
F4 param.msg
F5 rescue.msg
F7 snake.msg
label linux
  kernel vmlinuz
  append initrd=initrd.img
label text
  kernel vmlinuz
  append initrd=initrd.img text
label expert
  kernel vmlinuz
  append expert initrd=initrd.img
label ks
  kernel vmlinuz
  append ks initrd=initrd.img
label lowres
  kernel vmlinuz
  append initrd=initrd.img lowres
```

5. Insert the following line in the beginning of the file, so that it is the first line.

```
serial 0 115200
```

6. Locate the line "default <my_label>". Usually "default linux" appears.

7. Locate the line "label <my_label>". Usually "label linux" appears. Under this line, the following appears:

```
kernel ...
append ...
```

8. Add the following parameters to the "append" line of <my_label>:

```
text console=ttyS0,115200
```

Note: In the above string, "ttyS0,115200" consists of a capital "S", only zeros and one comma.

The "kernel" and " "append" lines should look like the following example:

```
label linux
  kernel vmlinuz
  append initrd=initrd.img ramdisk_size=8192
     text console=ttyS0,115200
```

9. Locate the line "prompt <flag>". Usually "prompt 1" appears. Change it to "prompt 0".

10. Locate the line "timeout <tenth_of_secs>". Usually "timeout 600" appears. Change it to "timeout 0".

Note: If the timeout line does not exist, **do not** add it.

11. The **isolinux.cfg** file should now look like the following:

```
serial 0 115200
default linux
prompt 0
...
label linux
  kernel vmlinuz
  append initrd=initrd.img ramdisk_size=8192
     text console=ttyS0,115200
```

12. Save the changes to the **isolinux.cfg** file.

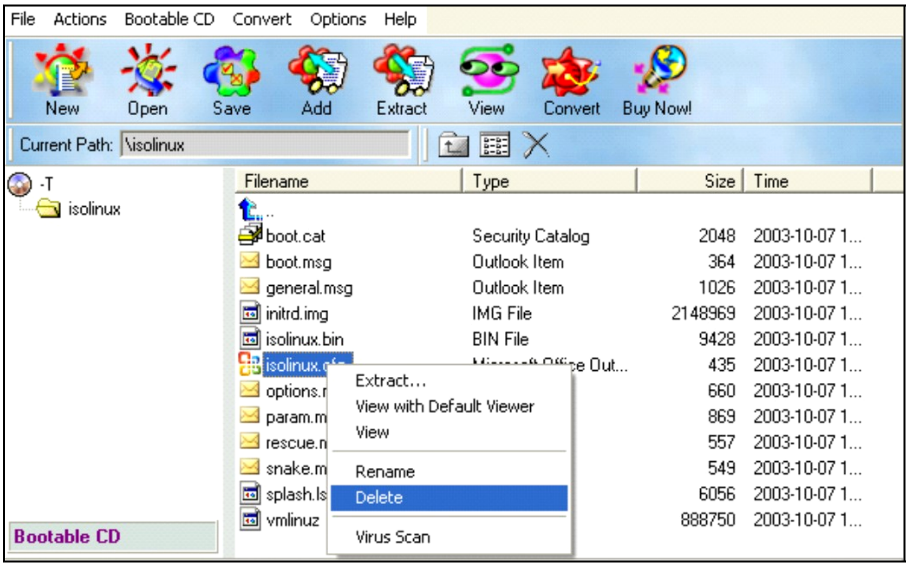
13. Close the text editor.

14. Open the **Partner Install** folder and with the ISO editing utility, open the **boot.iso** file.

15. Double-click the **isolinux** directory. The folder contents opens.

16. Right-click the **isolinux.cfg** file, and then from the shortcut menu, choose **Delete** to delete this file.

Figure L-9: Deleting CFG

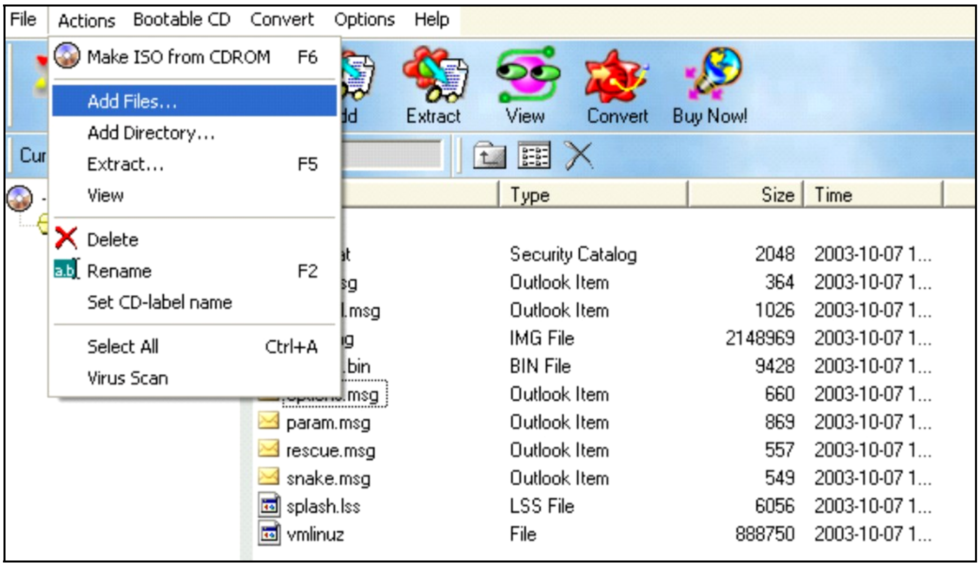


Current Path: \isolinux

Filename	Type	Size	Time
..			
boot.cat	Security Catalog	2048	2003-10-07 1...
boot.msg	Outlook Item	364	2003-10-07 1...
general.msg	Outlook Item	1026	2003-10-07 1...
initrd.img	IMG File	2148969	2003-10-07 1...
isolinux.bin	BIN File	9428	2003-10-07 1...
isolinux.cfg	Microsoft Office Out...	435	2003-10-07 1...
options.r		660	2003-10-07 1...
param.m		869	2003-10-07 1...
rescue.n		557	2003-10-07 1...
snake.m		549	2003-10-07 1...
splash.ls		6056	2003-10-07 1...
vmlinuz		888750	2003-10-07 1...

17. From the ISO edit utility menu, select the **Actions** option, followed by **Add Files**.

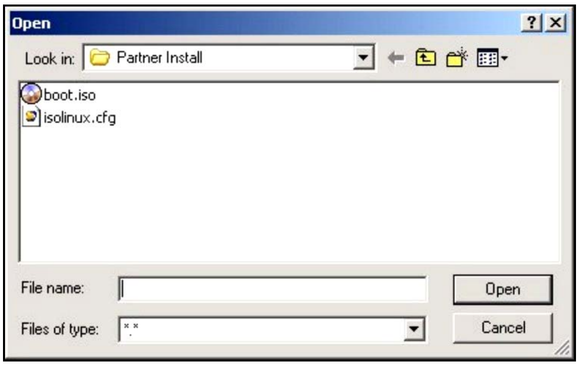
Figure L-10: File Add



Filename	Type	Size	Time
t	Security Catalog	2048	2003-10-07 1...
sg	Outlook Item	364	2003-10-07 1...
l.msg	Outlook Item	1026	2003-10-07 1...
g	IMG File	2148969	2003-10-07 1...
.bin	BIN File	9428	2003-10-07 1...
.msg	Outlook Item	660	2003-10-07 1...
param.msg	Outlook Item	869	2003-10-07 1...
rescue.msg	Outlook Item	557	2003-10-07 1...
snake.msg	Outlook Item	549	2003-10-07 1...
splash.lss	LSS File	6056	2003-10-07 1...
vmlinuz	File	888750	2003-10-07 1...

18. Navigate to the **Partner Install** folder, select the **isolinux.cfg** file, and then click **Open**.

Figure L-11: ISO Open Function



The updated **isolinux.cfg** file has now been copied from the **Partner Install** folder to the **boot.iso** image.

19. Save the **boot.iso** image in the **Partner Install** folder.

L.5.3 Stage 3: Burn the CD

➢ To burn the CD, take these 3 steps:

1. Open a burning utility.
2. Use the **Burn Image** option to burn the **boot.iso** to an empty CD media.
3. Mark the media as the “Boot CD”.

⚠ **Note:** Ensure that the **boot.iso** file should be burned as an image and not as a data file.

L.5.4 Stage 4: Installation

Now you have the boot media which enables the installation of the Mediant 1000 using the serial connection (terminal) with RS-232 cable.

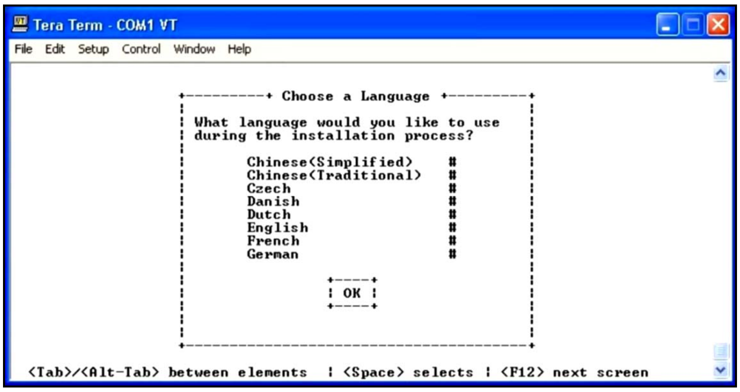
➢ To complete the installation, take these 9 steps:

1. Connect your Windows™ PC via serial cable to the Mediant 1000.
2. Connect the USB CD ROM device via USB cable to the Mediant 1000.
3. Power up the PC.
4. Open the Terminal application (e.g. HyperTerminal) on your Windows™ PC. Create a new connection with the following settings:
   - Connect Port = COM1
   - Baudrate = 115200 (bits per second)
   - Data Bits = 8
   - Parity = None
   - Stop Bits = 1
   - Flow Control = None
5. Insert the “Boot CD” (created in Stage 3) into the USB CD-ROM.
6. Power up the Mediant 1000.
7. On the Terminal application, the BIOS phase starts and the LinuxTM installation begins. The installation uncompresses the kernel, loads it and its drivers and starts the interactive installation.

> **Note:** After the BIOS phase, some badly formatted text may appear on the screen.

8. The first interactive screen should be 'Choose a Language'. Select the language you wish to use.

Figure L-12: Choose a Language



```
+---------+ Choose a Language +---------+
¦                                       ¦
¦ What language would you like to use   ¦
¦ during the installation process?      ¦
¦                                       ¦
¦       Chinese(Simplified)      #      ¦
¦       Chinese(Traditional)     #      ¦
¦       Czech                    #      ¦
¦       Danish                   #      ¦
¦       Dutch                    #      ¦
¦       English                  #      ¦
¦       French                   #      ¦
¦       German                   #      ¦
¦                                       ¦
¦                +----+                 ¦
¦                ¦ OK ¦                 ¦
¦                +----+                 ¦
¦                                       ¦
¦                                       ¦
+---------------------------------------+

 <Tab>/<Alt-Tab> between elements  ¦ <Space> selects ¦ <F12> next screen
```

9. When prompted for the installation media, **remove the “Boot CD” and insert the firts installation disk** and select 'CDROM' as your installation media. From this point on, you should proceed with the screen instructions, as instructed by your Linux^{TM} distributor.

L.5.5 Additional RedHat™ and Fedora™ Installation Notes (7)

Please refer to the following notes for the remaining part of the installation.

1. Select **LILO** as your bootloader where possible, otherwise select GRUB.
2. It is recommended that you disable the firewall when prompted (select "**No Firewall**").
3. In case you forget to disable the firewall during the installation and want to do it after the installation, run the following command:

   ```
   /usr/bin/redhat-config-securitylevel-tui --quiet -disabled
   ```
4. It is recommended that you assign a **static IP address** to your Mediant 1000. So when the installation has been completed, you will be able to create an SSH remote connection and continue the post-installation configuration.
5. During the bootloader configuration (after you selected which bootloader you want to install), you will be prompted to provide additional text to be appended to the kernel. Ensure that the installation (grub-installer or lilo-installer) recognizes the serial console and contains the following text:

   ```
   console=ttyS0,115200
   ```
6. If there is no text (i.e. the installer did not recognize the serial console), then insert the following:

   ```
   text console=ttyS0,115200
   ```
7. Once the installation has been completed, you will be prompted to re-start the Mediant 1000.

Note: If the installed kernel is version 2.6 or later (Fedora Core 4 or later, RedHat Enterprise 4 or later) then refer to the Post-installation Notes for Kernels 2.6+, below.

L.5.6 Post-installation Notes for Kernels 2.6+ (Fedora™ Core 4+ and RedHat™ EL 4+): (3 Notes)

1. When the Mediant 1000 is re-booting and after the BIOS phase, there is a bootloader phase (GRUB or LILO) which starts uncompressing and loading the kernel. After the kernel is loaded the services will start. While the services are loading and the message "**Press 'I' to enter interactive startup**" appears, press "I".
2. Once the hardware has been detected, you will be prompted whether to start each service. For the **syslog service**, select "N" (in order NOT to load it).
3. After you login (either serially or using SSH) to the Mediant 1000, you should disable the syslog service from being started during system startup. To do this, you have to disable the **S**syslog** file located in both /etc/rc3.d and in /etc/rc5.d directories. The file is usually called **S12syslog**.
4. Rename it to K12syslog with the following command.

   ```
   mv S12syslog K12syslog
   ```

L.6 Linux™ Debian Installation Instructions

Perform the following five stages for the Linux™ Debian Installation. (Some Debian distributions may vary slightly).

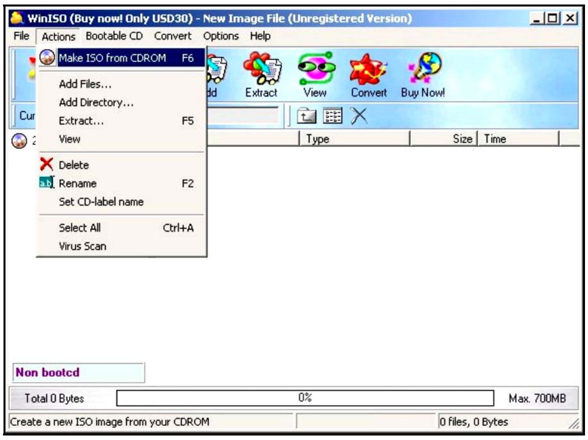
L.6.1 Stage 1: Obtaining the ISO Image

To obtain an updated ISO image, create it using the steps detailed in the section below.

➢ To create an ISO image using an ISO editor utility, take these 5 steps:

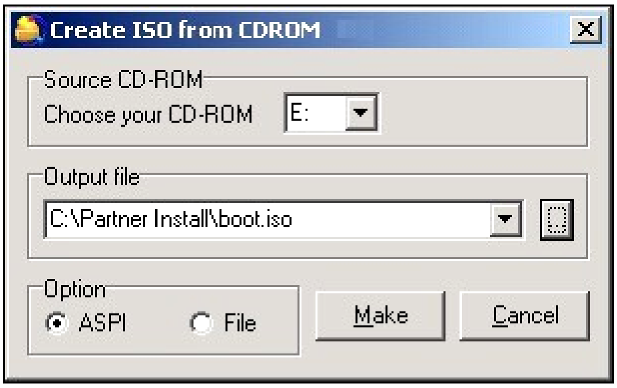
1. Insert the 1st installation disk of the Linux™ Debian distribution into the CD-ROM drive of the Windows™ PC.
2. Using Internet Explorer, download a utility that allows editing of an ISO image. The WinISO™ program (http://www.winiso.com/download.htm) may be used.
3. Start WinISO™, and then from the Actions menu, choose **Make ISO from CDROM**.

Figure L-13: WinISO - Actions Screen



4. Create a **Partner Install** folder on your hard drive. Select **boot.iso** as the output filename, and then click **Make**.

Figure L-14: Create ISO from CD-ROM



Source CD-ROM

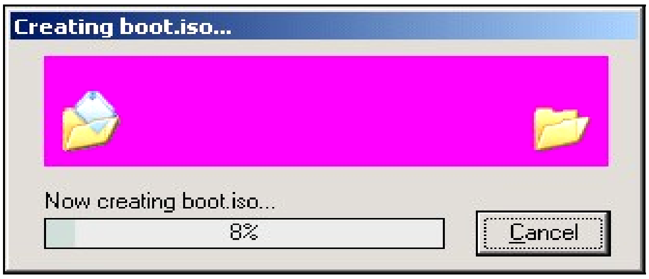
Choose your CD-ROM: E:

Output file: C:\Partner Install\boot.iso

Option: ● ASPI ○ File

The **.iso** file starts being created.

Figure L-15: Creating .iso file



Creating boot.iso...

Now creating boot.iso...

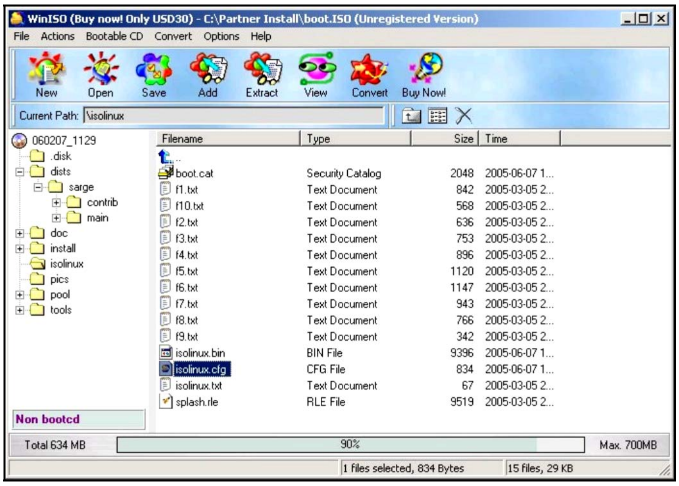
8%

L.6.2 Stage 2: Preparing the Boot Media

➢ To prepare the Boot Media, take these 5 steps:

1. If you have not already done so, download a utility that allows editing of an ISO image. The WinISO™ program (http://www.winiso.com/download.htm) may be used.
2. Using Internet Explorer, download a Unix File Format text editor. PSPad™ (http://www.pspad.com) or UltraEdit™ (http://www.ultraedit.com) may be used.
3. Locate the **boot.iso** file in the **Partner Install** folder on the hard disk of your PC and with the ISO image utility, navigate to the **\isolinux\isolinux.cfg** file.

Figure L-16: Partner Install folder



WinISO (Buy now! Only USD30) - C:\Partner Install\boot.ISO (Unregistered Version)

Current Path: \isolinux

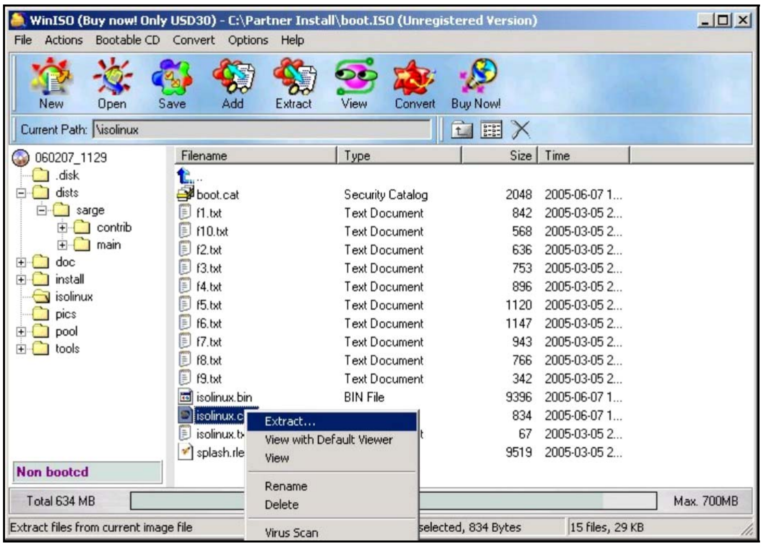
Filename	Type	Size	Time
..			
boot.cat	Security Catalog	2048	2005-06-07 1...
f1.txt	Text Document	842	2005-03-05 2...
f10.txt	Text Document	568	2005-03-05 2...
f2.txt	Text Document	636	2005-03-05 2...
f3.txt	Text Document	753	2005-03-05 2...
f4.txt	Text Document	896	2005-03-05 2...
f5.txt	Text Document	1120	2005-03-05 2...
f6.txt	Text Document	1147	2005-03-05 2...
f7.txt	Text Document	943	2005-03-05 2...
f8.txt	Text Document	766	2005-03-05 2...
f9.txt	Text Document	342	2005-03-05 2...
isolinux.bin	BIN File	9396	2005-06-07 1...
isolinux.cfg	CFG File	834	2005-06-07 1...
isolinux.txt	Text Document	67	2005-03-05 2...
splash.rle	RLE File	9519	2005-03-05 2...

Total 634 MB 90% Max. 700MB

1 files selected, 834 Bytes 15 files, 29 KB

4. Extract the **isolinux.cfg** file by right-clicking the file name, and then from the shortcut menu, choosing **Extract**.

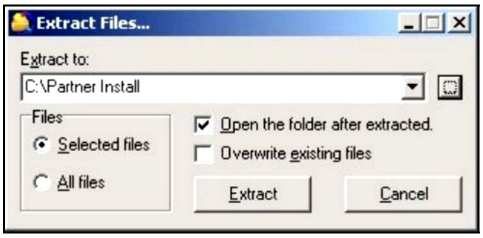
Figure L-17: Extract isolinux.cfg



Filename	Type	Size	Time
..			
boot.cat	Security Catalog	2048	2005-06-07 1...
f1.txt	Text Document	842	2005-03-05 2...
f10.txt	Text Document	568	2005-03-05 2...
f2.txt	Text Document	636	2005-03-05 2...
f3.txt	Text Document	753	2005-03-05 2...
f4.txt	Text Document	896	2005-03-05 2...
f5.txt	Text Document	1120	2005-03-05 2...
f6.txt	Text Document	1147	2005-03-05 2...
f7.txt	Text Document	943	2005-03-05 2...
f8.txt	Text Document	766	2005-03-05 2...
f9.txt	Text Document	342	2005-03-05 2...
isolinux.bin	BIN File	9396	2005-06-07 1...
isolinux.c		834	2005-06-07 1...
isolinux.t		67	2005-03-05 2...
splash.rle		9519	2005-03-05 2...

5. Extract the **isolinux.cfg** file to the **Partner Install** folder.

Figure L-18: Extracting to Partner Install folder



Extract to: C:\Partner Install

Files

● Selected files

○ All files

☑ Open the folder after extracted.

□ Overwrite existing files

L.6.3 Stage 3: Editing the isolinux.cfg File

In order to obtain an updated isolinux.cfg file, either download it from the AudioCodes website as described in Section L.6.3.1, or edit it using the steps detailed in Section L.6.3.2.

L.6.3.1 Downloading an updated Debian isolinux.cfg file

➢ **To download an updated Debian isolinux.cfg file from the AudioCodes website, take these 6 steps:**

1. From http://www.audiocodes.com, navigate to the Support page.
2. Login with your Username and Password.
3. Under **Product Documentation**, select the **Mediant 1000** link and click **Mediant 1000 OSN Server**.
4. Select **Linux Boot Image** and select the installation you need.
5. Download the **isolinux.cfg** compressed file to a folder called **Partner Install** on your PC and extract it.
6. Continue with Step 11 on page 486.

L.6.3.2 Editing the isolinux.cfg File

➢ To edit the isolinux.cfg file, take these 16 steps:

1. Open the **Partner Install** folder and select the **isolinux.cfg** file, with a text editor that supports UNIX file format (e.g., PSPad, UltraEdit).

2. Insert the following line in the beginning of the file, so that it is the first line.

   ```
   serial 0 115200
   ```

3. Locate the line 'default <my_kernel>'. It is usually the first line of the file and appears as follows.

   ```
   DEFAULT /install/vmlinuz
   ```

4. Under the **default** line there should be a line that begins with 'append'.

5. Add the following parameters to the 'append' line.

   ```
   text console=ttyS0,115200
   ```

Note: In the above string, “ttyS0,115200” consists of a captial “S”, only zeros and one comma.

The 'default <my_kernel> and 'append' lines should look like the following example:

```
DEFAULT /install/vmlinuz
APPEND vga=normal initrd=/install/initrd.gz text
       console=ttyS0,115200 ramdisk_size=10240 root=/dev/rd/0
       devfs=mount,dall rw –
```

6. Locate the line 'prompt <flag>'. It usually appears as 'prompt 1. Change it to 'prompt 0'.

7. Locate the line 'timeout <tenth_of_secs>'. It usually appears as 'timeout 600'. Change it to 'timeout 0'.

Note: If the timeout line does not exist, **do not** add it.

8. The **isolinux.cfg** file should now look like the following:

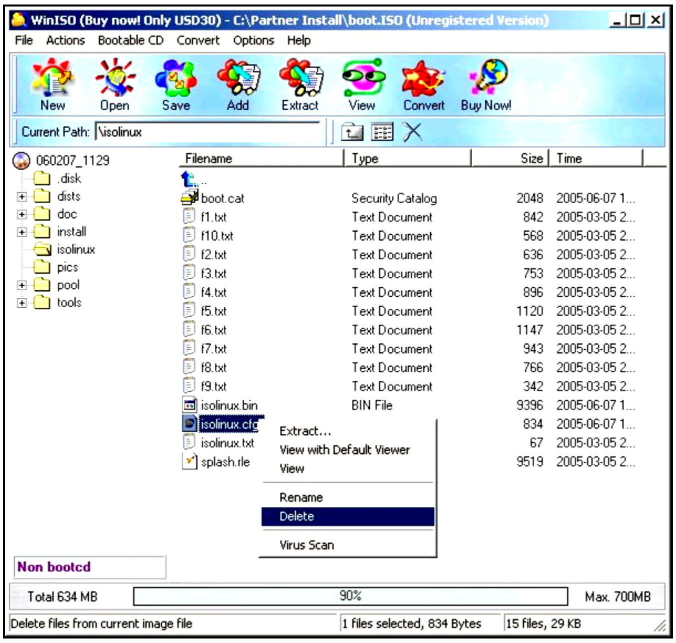
   ```
   serial 0 115200
   DEFAULT /install/vmlinuz

   APPEND  vga=normal initrd=/install/initrd.gz text
           console=ttyS0,115200 ramdisk_size=10240 root=/dev/rd/0
           devfs=mount,dall rw --

   LABEL linux
     kernel /install/vmlinuz
   ...
   DISPLAY isolinux.txt
   TIMEOUT 0
   PROMPT 0
   F1 f1.txt
   F2 f2.txt
   F3 f3.txt
   F4 f4.txt
   F5 f5.txt
   F6 f6.txt
   ```

```
F7 f7.txt
F8 f8.txt
F9 f9.txt
F0 f10.txt
```

Note: If the timeout line does not exist, do not add it.1. If you want to install 'kernel 2.6' rather than the 'default 2.4 version', then:
a) Take the options from 'kernel' and 'append' lines under the label called 'LABEL linux26'.
b) Replace the options of 'DEFAULT' and 'APPEND' lines (at the start of the file).
c) Apply the changes from Step 8.

9. Save the changes to the **isolinux.cfg** file.
10. Close the text editor.
11. Open the **Partner Install** folder and with the ISO editing utility, open the **boot.iso** file.
12. Click the **isolinux** directory. The folder contents opens.
13. Right-click the **isolinux.cfg** file, and then from the shortcut menu, choose **Delete** to delete this file.

Figure L-19: Deleting CFG



WinISO (Buy now! Only USD30) - C:\Partner Install\boot.ISO (Unregistered Version)

Current Path: \isolinux

Filename	Type	Size	Time
..			
boot.cat	Security Catalog	2048	2005-06-07 1...
f1.txt	Text Document	842	2005-03-05 2...
f10.txt	Text Document	568	2005-03-05 2...
f2.txt	Text Document	636	2005-03-05 2...
f3.txt	Text Document	753	2005-03-05 2...
f4.txt	Text Document	896	2005-03-05 2...
f5.txt	Text Document	1120	2005-03-05 2...
f6.txt	Text Document	1147	2005-03-05 2...
f7.txt	Text Document	943	2005-03-05 2...
f8.txt	Text Document	766	2005-03-05 2...
f9.txt	Text Document	342	2005-03-05 2...
isolinux.bin	BIN File	9396	2005-06-07 1...
isolinux.cfg		834	2005-06-07 1...
isolinux.txt		67	2005-03-05 2...
splash.rle		9519	2005-03-05 2...

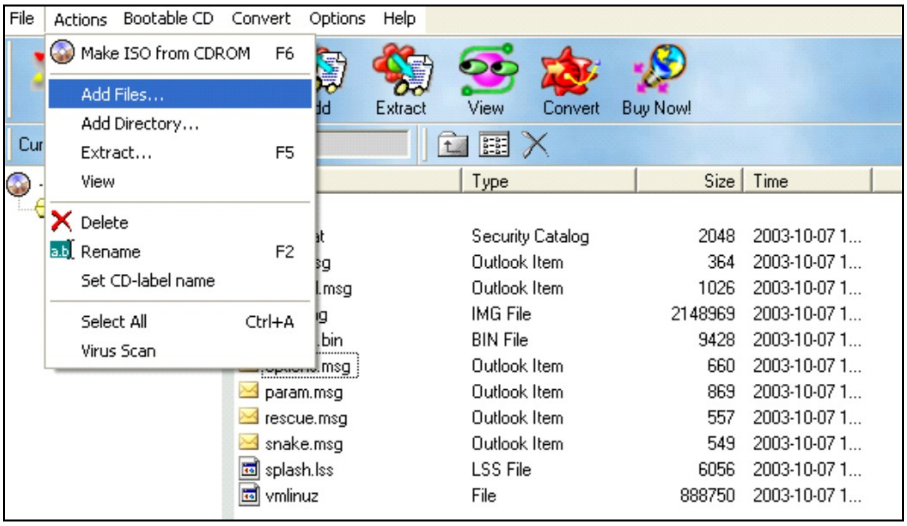
Non bootcd

Total 634 MB 90% Max. 700MB

Delete files from current image file | 1 files selected, 834 Bytes | 15 files, 29 KB

14. From the ISO edit utility menu, select the **Actions** option, followed by **Add Files**.

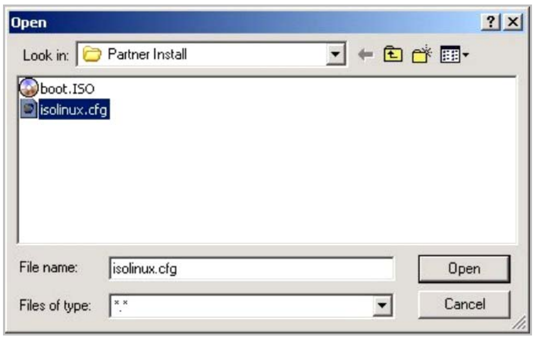
Figure L-20: File Add



	Type	Size	Time
t	Security Catalog	2048	2003-10-07 1...
sg	Outlook Item	364	2003-10-07 1...
.msg	Outlook Item	1026	2003-10-07 1...
g	IMG File	2148969	2003-10-07 1...
.bin	BIN File	9428	2003-10-07 1...
.msg	Outlook Item	660	2003-10-07 1...
param.msg	Outlook Item	869	2003-10-07 1...
rescue.msg	Outlook Item	557	2003-10-07 1...
snake.msg	Outlook Item	549	2003-10-07 1...
splash.lss	LSS File	6056	2003-10-07 1...
vmlinuz	File	888750	2003-10-07 1...

15. Navigate to the **Partner Install** folder, select the **isolinux.cfg** file, and then click **Open**.

Figure L-21: ISO Open Function



Look in: Partner Install

boot.ISO
isolinux.cfg

File name: isolinux.cfg

Files of type: *.*

The updated **isolinux.cfg** file is copied to the **Partner Install\isolinux** directory.

16. Save the **boot.iso** in the **Partner Install** folder.

L.6.4 Stage 4: Burn the CD

➢ To burn the CD, take these 3 steps:

1. Open a burning utility.
2. Use the **Burn Image** option to burn the boot.iso to an empty CD media.
3. Mark the media as the “Boot CD”.

Note: Ensure that the **boot.iso** file should be burned as an image and not as a data file.

L.6.5 Stage 5: Installation

Now you have the boot media which enables the installation of the Mediant 1000 using the serial connection (terminal) with RS-232 cable.

➢ To complete the installation, take these 7 steps:

1. Connect your Windows™ PC via serial cable to the Mediant 1000.
2. Connect the USB CDROM device via USB cable to the Mediant 1000.
3. Power up the PC.
4. Open the Terminal application on your PC. Create a new connection with the following settings:
   - Connect Port = COM1
   - Baudrate = 115200 (bits per second)
   - Data Bits = 8
   - Parity = None
   - Stop Bits = 1
   - Flow Control = None
5. Insert the “Boot CD” (created in Stage 3) into the USB CD-ROM.
6. Power up the Mediant 1000.
7. On the Terminal screen, the BIOS phase will start and the Linux^{TM} installation will begin. The installation will uncompress the kernel, load it and its drivers and start the interactive installation.

Notes:

- After the BIOS phase, some badly formatted text may appear on the screen.
- From this point on, you should proceed with the screen instructions, as instructed by your Linux distributor

L.6.6 Additional Linux™ Debian Installation Notes (12)

Please refer to the following notes for the remaining part of the installation.

1. The first interactive screen should be 'Language Selection'. **Before you select the language you wish to use, remove the “Boot CD” and insert the 1st Debian installation disk**. Select the language you wish to use.
2. As you continue with the installation, you should go through the following stages of the installation:
   a. Country selection
   b. CROM scan
   c. DHCP configuring
3. Configure your network settings manually by selecting the **Configure network manually** option. Either:
   - the DHCP configuration will fail and a screen enabling you to configure the network manually will appear,
     or
   - go to the Main Installation menu and select the manual network configuration. Ensure the network is connected and set a **Static IP Address** and all the other network parameters.

4. Continue with the installation, going through the following screens:
   - 'Partition disks'
   - 'Installing the Debian base system'
5. In the 'Install the GRUB boot loader on hard disk' screen, select **Go Back**. This returns you to the 'Debian installer main menu' screen. This is the main menu of the installation.
6. Scroll down in the menu and instead of using GRUB select **Install LILO boot loader on a hard disk**.
7. In the 'LILO Installation target', select **/dev/hda: Master Boot Record**.

Note: The bootloader should detect the serial console. If it doesn't then either.
a) pass additional parameters to the kernel (configure that using the main installation menu)
or/and
b) when configuring LILO you should specify that it tells the kernel to use serial console at speed 115200. In other words append "`text console=ttyS0,115200`" to the kernel options using LILO.
If the serial console wasn't detected during installation, add the following line to the LILO's configuration file after the installation:
"`serial=0,115200`"

8. In the next screen, the installation should display the following:

```
LILO is configured to use serial port ttyS0 as the console.
The serial port speed is set to 115200.
```

9. Select **Continue**.

   The 'Finish the installation' screen should appear followed by **'Installation complete'**.
10. The installation should eject the CD. Please remove the CD and close the tray.

    The system reboots and after the BIOS phase you should see the kernel being uncompressed.
11. Now the installation has finished its "base" phase. The 'Debian Configuration' screen should appear. This is the phase where you select additional packages to be installed and where they are going to be installed from.

Note: Ensure that you install **telnet** and that the **ssh** packages get installed, so that you can connect to your newly-installed system in the future (usually installed by default). From this point on you should proceed as instructed by your Linux distributor.

12. After the whole installation has been completed, you will be able login to the system from the serial console and/or to "ssh" on your Mediant 1000 (to create an SSH remote connection to it) and to continue its post-installation configuring. You can use the boot media you have created in order to install multiple Mediant 1000 stations.

L.7 Linux™ SUSE Installation Instructions

Perform the following five stages for the Linux^{TM} SUSE Installation. (Some SUSE distributions may vary slightly).

L.7.1 Additional Requirement for Linux™ SUSE Installation

In order to install Linux™ SUSE, a terminal emulation program is needed which supports:

- ANSI colors (or LinuxTM emulation)
- Changing terminal size (to 132x47).
  The Tera Term™ program may be used. Refer to
  http://hp.vector.co.jp/authors/VA002416/teraterm.html

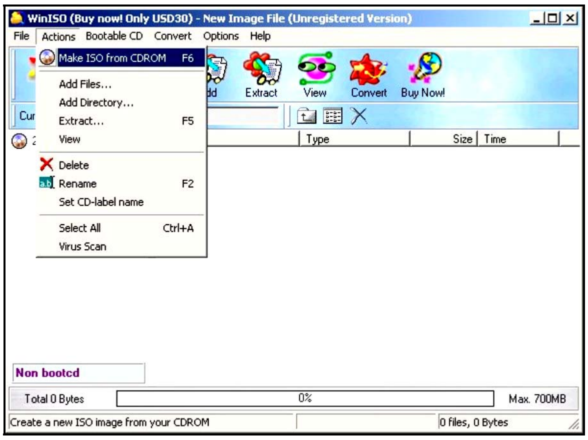
L.7.2 Stage 1: Obtaining the ISO Image

In order to obtain an updated ISO image, create it using the steps detailed in the section below.

➢ To create an ISO image using an ISO editor utility, take these 4 steps:

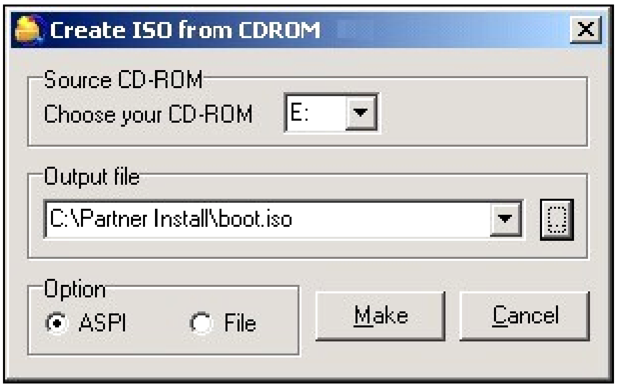
1. Insert the 1st installation disk of the Linux™ SUSE distribution into the CD-ROM drive of the Windows™ PC.
2. Using Internet Explorer, download a utility that allows editing of an ISO image. The WinISO™ program (http://www.winiso.com/download.htm) may be used.
3. Start WinISO™ and from the main menu, select **Actions – Make ISO from CDROM**.

Figure L-22: WinISO - Actions Screen



4. Create a **Partner Install** folder on your hard drive. Select **boot.iso** as the output filename, and then click **Make**.

Figure L-23: Create ISO from CD-ROM



Create ISO from CDROM

Source CD-ROM

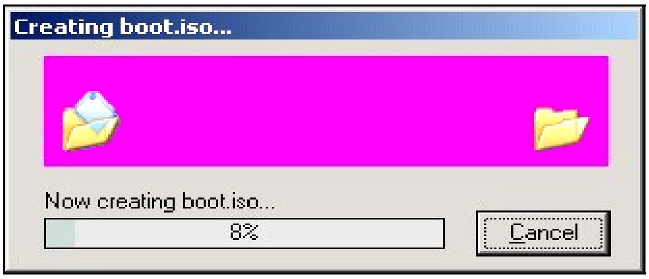
Choose your CD-ROM: E:

Output file: C:\Partner Install\boot.iso

Option: ● ASPI ○ File

The **.iso** file begins being created.

Figure L-24: Creating .iso file



Creating boot.iso...

Now creating boot.iso...

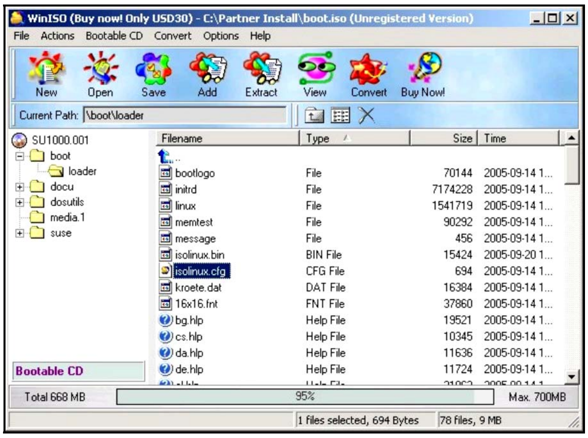
8%

L.7.3 Stage 2: Preparing the Boot Media

➢ To prepare the Boot Media, take these 5 steps:

1. If you have not already done so, download a utility that allows editing of an ISO image. The WinIso™ program (http://www.winiso.com/download.htm) may be used.
2. Using Internet Explorer, download a Unix File Format text editor. PSPad^{TM} (http://www.pspad.com) or UltraEdit^{TM} (http://www.ultraedit.com) may be used.
3. Locate the **boot.iso** file in the **Partner Install** folder on the hard disk of your PC and navigate to the **isolinux.cfg** file.

Figure L-25: Partner Install folder



WinISO (Buy now! Only USD30) - C:\Partner Install\boot.iso (Unregistered Version)

Current Path: \boot\loader

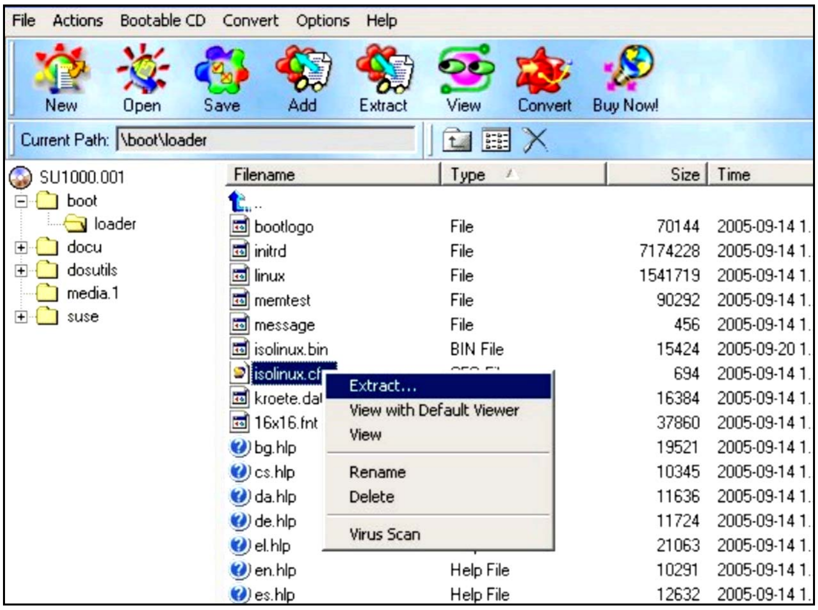
Filename	Type	Size	Time
..			
bootlogo	File	70144	2005-09-14 1...
initrd	File	7174228	2005-09-14 1...
linux	File	1541719	2005-09-14 1...
memtest	File	90292	2005-09-14 1...
message	File	456	2005-09-14 1...
isolinux.bin	BIN File	15424	2005-09-20 1...
isolinux.cfg	CFG File	694	2005-09-14 1...
kroete.dat	DAT File	16384	2005-09-14 1...
16x16.fnt	FNT File	37860	2005-09-14 1...
bg.hlp	Help File	19521	2005-09-14 1...
cs.hlp	Help File	10345	2005-09-14 1...
da.hlp	Help File	11636	2005-09-14 1...
de.hlp	Help File	11724	2005-09-14 1...

Total 668 MB 95% Max. 700MB

1 files selected, 694 Bytes 78 files, 9 MB

4. Extract the **isolinux.cfg** file by right-clicking the filename, and then from the shortcut menu, choosing **Extract**.

Figure L-26: Extract isolinux.cfg

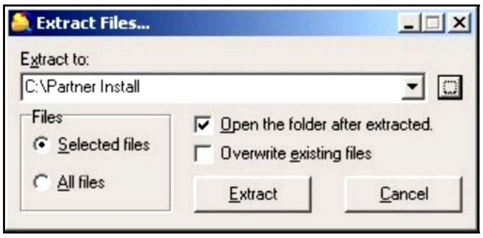


Current Path: \boot\loader

Filename	Type	Size	Time
..			
bootlogo	File	70144	2005-09-14 1
initrd	File	7174228	2005-09-14 1
linux	File	1541719	2005-09-14 1
memtest	File	90292	2005-09-14 1
message	File	456	2005-09-14 1
isolinux.bin	BIN File	15424	2005-09-20 1
isolinux.cf		694	2005-09-14 1
kroete.dat		16384	2005-09-14 1
16x16.fnt		37860	2005-09-14 1
bg.hlp		19521	2005-09-14 1
cs.hlp		10345	2005-09-14 1
da.hlp		11636	2005-09-14 1
de.hlp		11724	2005-09-14 1
el.hlp		21063	2005-09-14 1
en.hlp	Help File	10291	2005-09-14 1
es.hlp	Help File	12632	2005-09-14 1

5. Extract the **isolinux.cfg** file to the **Partner Install** folder.

Figure L-27: Extracting to Partner Install folder



Extract to: C:\Partner Install

Files

● Selected files

○ All files

☑ Open the folder after extracted.

□ Overwrite existing files

L.7.4 Stage 3: Editing the isolinux.cfg File

To obtain an updated isolinux.cfg file, either download it from the AudioCodes Web site as described in Section L.7.4.1, or edit it using the steps detailed in Section L.7.4.2.

L.7.4.1 Downloading an updated SUSE isolinux.cfg file

➢ **To download an updated SUSE isolinux.cfg file from the AudioCodes website, take these 6 steps:**

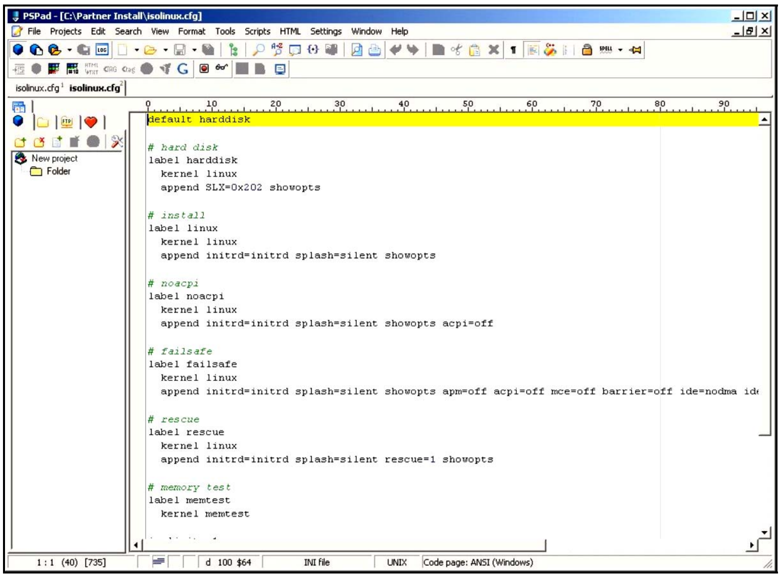
1. From http://www.audiocodes.com, navigate to the Support page.
2. Login with your Username and Password.
3. Under **Product Documentation**, select the **Mediant 1000** link and click **Mediant 1000 OSN Server**.
4. Select **Linux Boot Image** and select the installation you need.
5. Download the **isolinux.cfg** compressed file to a folder called **Partner Install** on your PC and extract it.
6. Continue with Step 14 on page 495.

L.7.4.2 Editing the isolinux.cfg File

➢ To edit the isolinux.cfg file, take these 19 steps:

1. From the **Partner Install** folder, open the **isolinux.cfg** file with a text editor that supports UNIX file format (e.g., PSPad, UltraEdit).

Figure L-28: isolinux.cfg


PSPad - [C:\Partner Install\isolinux.cfg]
File Projects Edit Search View Format Tools Scripts HTML Settings Window Help
isolinux.cfg isolinux.cfg
New project
Folder
1 : 1 (40) [735]
d 100 $64
INI file
UNIX
Code page: ANSI (Windows)

```
default harddisk

# hard disk
label harddisk
  kernel linux
  append SLX=0x202 showopts

# install
label linux
  kernel linux
  append initrd=initrd splash=silent showopts

# noacpi
label noacpi
  kernel linux
  append initrd=initrd splash=silent showopts acpi=off

# failsafe
label failsafe
  kernel linux
  append initrd=initrd splash=silent showopts apm=off acpi=off mce=off barrier=off ide=nodma id

# rescue
label rescue
  kernel linux
  append initrd=initrd splash=silent rescue=1 showopts

# memory test
label memtest
  kernel memtest
```

2. Insert the following line in the beginning of the file, so that it is the first line.

```
serial 0 115200
```

3. Locate the line 'DEFAULT <my_label>'. It is usually the first line of the file. It is usually 'default harddisk' or 'default linux'.
4. If <my_label> isn't 'linux' (for instance it can be 'harddisk'), change it to 'linux'.
5. Locate the following line: '**label linux**'.
6. Under the '**label linux**' line, two lines should appear.

```
kernel...
append...
```

The following examples show how the 'label linux' line and its kernel and append sublines could look like, before you change them:

```
label linux
kernel linux
      append initrd=initrd splash=silent showopts
or
label linux
kernel linux
      append initrd=initrd ramdisk_size=65536 splash=silent
       showopts
```

7. Remove the 'splash=...' and 'showopts' parameters, if they appear in the 'append' line.

Note: Do not remove any other parameters in the 'append' line, especially the 'initrd=' parameter.

8. Add the following parameters to the 'append' line: text console=ttyS0,115200.

Note: In the above string, 'ttyS0,115200' consists of a capital 'S', only zeros and one comma.

9. The following examples show how the '**label linux**' line and its kernel & append sublines could look like, after you change them::

```
label linux
  kernel linux
  append initrd=initrd text console=ttyS0,115200

or

label linux
  kernel linux
  append initrd=initrd ramdisk_size=65536 text
console=ttyS0,115200
```

10. Locate the line: 'prompt <flag>'. It usually appears as 'prompt 1'. Change it to 'prompt 0'.

11. Locate the line: 'timeout <tenth_of_secs>'. It usually appears as 'timeout 200'. Change it to 'timeout 0'.

Note: If the timeout line does not exist, **do not** add it.

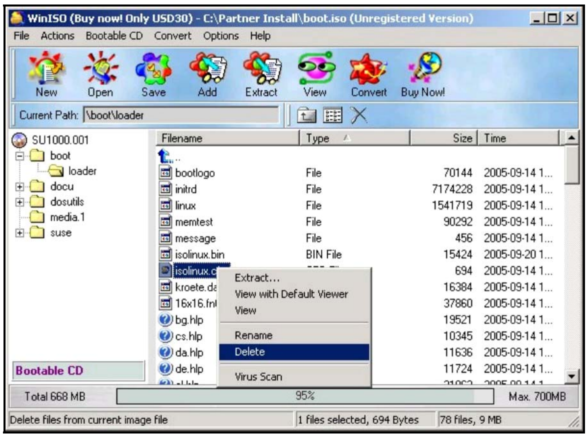
12. The '**isolinux.cfg**' file should now look like the following:

```
serial 0 115200
default linux

# hard disk
label harddisk
  kernel linux
  append SLX=0x202 showopts

# install
label linux
  kernel linux
  append initrd=initrd text console=ttyS0,115200

# noacpi
label noacpi
  kernel linux
  append initrd=initrd splash=silent showopts acpi=off

# failsafe
label failsafe
  kernel linux
  append initrd=initrd splash=silent showopts apm=off acpi=off
         mce=off barrier=off ide=nodma idewait=50 i8042.nomux
         psmouse.proto=bare irqpoll
```

```
# rescue
label rescue
  kernel linux
  append initrd=initrd splash=silent rescue=1 showopts

# memory test
label memtest
  kernel memtest

implicit  1
gfxboot      bootlogo
display      message
prompt       0
timeout      0
readinfo  2
framebuffer  1
notice       2
```

13. Save the changes to the **isolinux.cfg** file.
14. Close the text editor.
15. Open the **Partner Install** folder and with the ISO edit utility, open the **boot.iso** file.
16. Navigate to the **isolinux.cfg** file, right-click it, and then from the shortcut menu, choose **Delete** to delete this file.

Figure L-29: Deleting CFG


WinISO (Buy now! Only USD30) - C:\Partner Install\boot.iso (Unregistered Version)
File Actions Bootable CD Convert Options Help
New Open Save Add Extract View Convert Buy Now!
SU1000.001
boot
loader
docu
dosutils
media.1
suse
Extract...
View with Default Viewer
View
Rename
Delete
Virus Scan
Bootable CD
Total 668 MB
95%
Max. 700MB

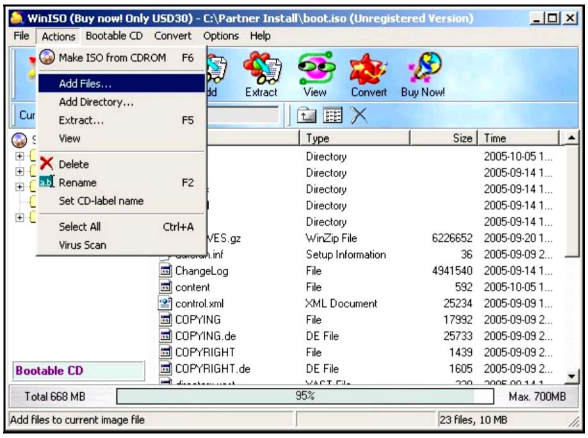
Current Path: \boot\loader

Filename	Type	Size	Time
..			
bootlogo	File	70144	2005-09-14 1...
initrd	File	7174228	2005-09-14 1...
linux	File	1541719	2005-09-14 1...
memtest	File	90292	2005-09-14 1...
message	File	456	2005-09-14 1...
isolinux.bin	BIN File	15424	2005-09-20 1...
isolinux.c		694	2005-09-14 1...
kroete.da		16384	2005-09-14 1...
16x16.fnt		37860	2005-09-14 1...
bg.hlp		19521	2005-09-14 1...
cs.hlp		10345	2005-09-14 1...
da.hlp		11636	2005-09-14 1...
de.hlp		11724	2005-09-14 1...

Delete files from current image file | 1 files selected, 694 Bytes | 78 files, 9 MB

17. From the ISO edit utility menu, select the **Actions** option, followed by **Add Files**.

Figure L-30: Add CFG File

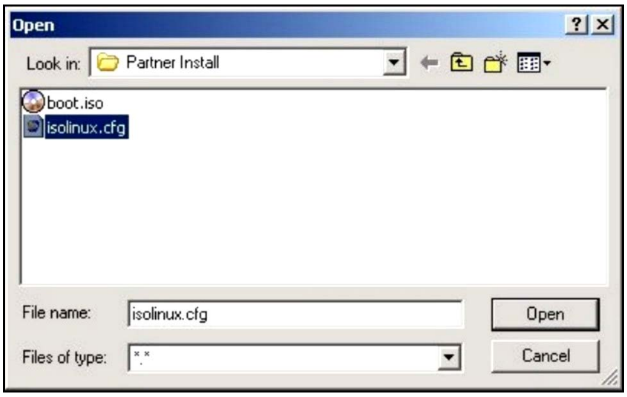


	Type	Size	Time
	Directory		2005-10-05 1...
	Directory		2005-09-14 1...
	Directory		2005-09-14 1...
	Directory		2005-09-14 1...
	Directory		2005-09-14 1...
VES.gz	WinZip File	6226652	2005-09-20 1...
.inf	Setup Information	36	2005-09-09 2...
ChangeLog	File	4941540	2005-09-14 1...
content	File	592	2005-10-05 1...
control.xml	XML Document	25234	2005-09-09 1...
COPYING	File	17992	2005-09-09 2...
COPYING.de	DE File	25733	2005-09-09 2...
COPYRIGHT	File	1439	2005-09-09 2...
COPYRIGHT.de	DE File	1605	2005-09-09 2...

Add files to current image file | | 23 files, 10 MB

18. Navigate to the **Partner Install** folder, select the **isolinux.cfg** file, and then click **Open**.

Figure L-31: Partner Install Folder



Look in: Partner Install

boot.iso
isolinux.cfg

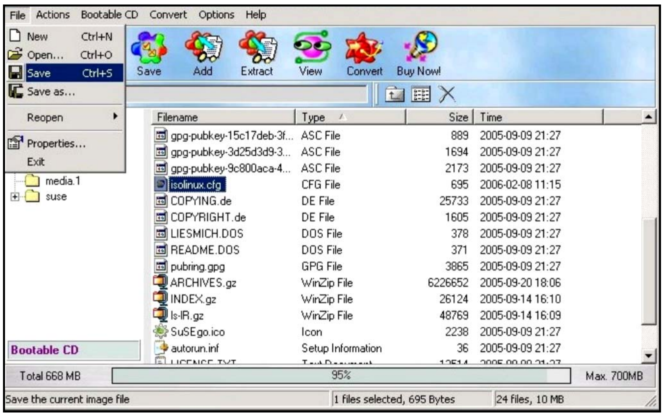
File name: isolinux.cfg

Files of type: *.*

The updated **isolinux.cfg** file is added to the **Partner Install** folder.

19. Save the **boot.iso** in the **Partner Install** folder.

Figure L-32: Save boot.iso



Filename	Type	Size	Time
gpg-pubkey-15c17deb-3f...	ASC File	889	2005-09-09 21:27
gpg-pubkey-3d25d3d9-3...	ASC File	1694	2005-09-09 21:27
gpg-pubkey-9c800aca-4...	ASC File	2173	2005-09-09 21:27
isolinux.cfg	CFG File	695	2006-02-08 11:15
COPYING.de	DE File	25733	2005-09-09 21:27
COPYRIGHT.de	DE File	1605	2005-09-09 21:27
LIESMICH.DOS	DOS File	378	2005-09-09 21:27
README.DOS	DOS File	371	2005-09-09 21:27
pubring.gpg	GPG File	3865	2005-09-09 21:27
ARCHIVES.gz	WinZip File	6226652	2005-09-20 18:06
INDEX.gz	WinZip File	26124	2005-09-14 16:10
ls-lR.gz	WinZip File	48769	2005-09-14 16:09
SuSEgo.ico	Icon	2238	2005-09-09 21:27
autorun.inf	Setup Information	36	2005-09-09 21:27
LICENSE.TXT	Text Document	12514	2005-09-09 21:27

Total 668 MB 95% Max. 700MB

Save the current image file | 1 files selected, 695 Bytes | 24 files, 10 MB

L.7.5 Stage 4: Burn the CD

➢ To burn the CD image, take these 3 steps:

1. Open a burning utility.
2. Use the **Burn Image** option to burn the **boot.iso** to an empty CD media.
3. Mark the media as the "Boot CD".

Note: Ensure that the boot.iso file should be burned as an image and not as a data file.

L.7.6 Stage 5: Installation

Now you have the boot media which enables SUSE installation of the Mediant 1000 using serial connection (terminal) with RS232 cable.

➢ To complete the installation, take these 8 steps:

1. Connect your Windows™ PC via serial cable to the Mediant 1000.
2. Open the Terminal application on your Windows™ PC. (Refer to the Additional Requirement in Section L.7.1 on page 490) Create a new connection with the following:
    - Connect Port = COM1
    - Baudrate = 115200 (or bits per second)
    - Data Bits = 8
    - Parity = None
    - Stop Bits = 1
    - Flow Control = None
    - Terminal settings: = 132x47, ANSI Color (optional)

> **Note:** If you are using Tera Term, navigate to the "Setup" option in the main menu and use the following configuration.
> Serial port:
>   Baud rate: 115200
> Terminal Setup:
>   Terminal size: 132x47
>   Term size = win size
>   Terminal ID: VT100
> Window:
>   Scroll buffer: 47
> Font:
>   Font: Terminal
>   Size: 8 (the default size is 10. If the window "slides" beyond your display then set the size to 8)

3. Connect the USB CD-ROM device via USB cable to the M1K.
4. Insert the "Boot CD" to the USB CD-ROM.

5. Power up the Mediant 1000.
6. On the Terminal screen, the BIOS phase will start and the Linux™ installation will begin. The installation will uncompress the kernel, load it and its drivers and start the interactive installation.

> **Note:** After the BIOS phase, some badly formatted text may appear on the screen.

> **Note:** From this point on, you should proceed with the screen instructions, as instructed by your Linux distributor.

7. During the installation process, the system may reboot at some stage. In this case, you will need to remove the CD from the CD ROM device.
8. During the installation of the 5th CD , the following items should be configured:
   - Username and password.
   - Firewall configuration: In the 'firewall selection/configuration' screen, select **Firewall: disabled**. In some distributions, you may have to select the **Change** option and set the configuration to 'manual'. This disables the firewall.
   - Network interfaces: The installation sets your network configuration to DHCP by default. If you plan to connect to the system immediately after the installation you’ll either have to know the IP address assigned to you by the DHCP or set a **static IP address, subnet mask and default gateway**.

After the whole installation has been completed, you will be able login to the system from the serial console and/or to 'ssh' on your Mediant 1000 (to create an SSH remote connection to it) and to continue its post-installation configuring. You can use the boot media you have created to install multiple Mediant 1000 stations.

M Regulatory Information (Digital Interface)

Declaration of Conformity

Application of Council Directives: 73/23/EEC (including amendments)
89/336/EEC (including amendments)
1999/5/EC Annex-II of the Directive

Standards to which Conformity is Declared: EN55022: 1998 + A1: 2000 + A2: 2003
EN55024:1998 + A1: 2001 + A2: 2003
EN61000-3-2: 2000 + A2: 2005
EN61000-3-3: 1995 + A1: 2001
EN60950-1: 2001

Manufacturer’s Name: AudioCodes Ltd.

Manufacturer’s Address: 1 Hayarden Street, Airport City, Lod 70151, Israel.

Type of Equipment: Digital VoIP System

Model Numbers: **Mediant 1000**

I, the undersigned, hereby declare that the equipment specified above conforms to the above Directives and Standards.

	27th June, 2006	Airport City, Lod, Israel
Signature	*Date (Day/Month/Year)*	*Location*

I. Zusmanovich, Compliance Engineering Manager

Czech	[AudioCodes Ltd] tímto prohlašuje, že tento [Mediant 1000 Digital Series] je ve shodě se základními požadavky a dalšími příslušnými ustanoveními směrnice 1999/5/ES."
Danish	Undertegnede [AudioCodes Ltd] erklærer herved, at følgende udstyr [Mediant 1000 Digital Series] overholder de væsentlige krav og øvrige relevante krav i direktiv 1999/5/EF
Dutch	Hierbij verklaart [AudioCodes Ltd] dat het toestel [Mediant 1000 Digital Series] in overeenstemming is met de essentiële eisen en de andere relevante bepalingen van richtlijn 1999/5/EG
English	Hereby, [AudioCodes Ltd], declares that this [Mediant 1000 Digital Series] is in compliance with the essential requirements and other relevant provisions of Directive 1999/5/EC.
Estonian	Käesolevaga kinnitab [AudioCodes Ltd] seadme [Mediant 1000 Digital Series] vastavust direktiivi 1999/5/EÜ põhinõuetele ja nimetatud direktiivist tulenevatele teistele asjakohastele sätetele.
Finnish	[AudioCodes Ltd] vakuuttaa täten että [Mediant 1000 Digital Series] tyyppinen laite on direktiivin 1999/5/EY oleellisten vaatimusten ja sitä koskevien direktiivin muiden ehtojen mukainen.
French	Par la présente [AudioCodes Ltd] déclare que l'appareil [Mediant 1000 Digital Series] est conforme aux exigences essentielles et aux autres dispositions pertinentes de la directive 1999/5/CE
German	Hiermit erklärt [AudioCodes Ltd], dass sich dieser/diese/dieses [Mediant 1000 Digital Series] in Übereinstimmung mit den grundlegenden Anforderungen und den anderen relevanten Vorschriften der Richtlinie 1999/5/EG befindet".
Greek	ΜΕ ΤΗΝ ΠΑΡΟΥΣΑ [AudioCodes Ltd] ΔΗΛΩΝΕΙ ΟΤΙ [Mediant 1000 Digital Series] ΣΥΜΜΟΡΦΩΝΕΤΑΙ ΠΡΟΣ ΤΙΣ ΟΥΣΙΩΔΕΙΣ ΑΠΑΙΤΗΣΕΙΣ ΚΑΙ ΤΙΣ ΛΟΙΠΕΣ ΣΧΕΤΙΚΕΣ ΔΙΑΤΑΞΕΙΣ ΤΗΣ ΟΔΗΓΙΑΣ 1999/5/ΕΚ
Hungarian	Alulírott, [AudioCodes Ltd] nyilatkozom, hogy a [Mediant 1000 Digital Series] megfelel a vonatkozó alapvetõ követelményeknek és az 1999/5/EC irányelv egyéb elõírásainak
Icelandic	æki þetta er í samræmi við tilskipun Evrópusambandsins 1999/5
Italian	Con la presente [AudioCodes Ltd] dichiara che questo [Mediant 1000 Digital Series] è conforme ai requisiti essenziali ed alle altre disposizioni pertinenti stabilite dalla direttiva 1999/5/CE.
Latvian	Ar šo [AudioCodes Ltd] deklarē, ka [Mediant 1000 Digital Series] atbilst Direktīvas 1999/5/EK būtiskajām prasībām un citiem ar to saistītajiem noteikumiem.
Lithuanian	[AudioCodes Ltd] deklaruoja, kad irenginys [Mediant 1000 Digital Series] tenkina 1999/5/EB Direktyvos esminius reikalavimus ir kitas sios direktyvos nuostatas
Maltese	Hawnhekk, [AudioCodes Ltd], jiddikjara li dan [Mediant 1000 Digital Series] jikkonforma mal-ħtiġijiet essenzjali u ma provvedimenti oħrajn relevanti li hemm fid-Dirrettiva 1999/5/EC
Norwegian	Dette produktet er i samhørighet med det Europeiske Direktiv 1999/5
Polish	[AudioCodes Ltd], deklarujemy z pelna odpowiedzialnoscia, ze wyrób [Mediant 1000 Digital Series] spelnia podstawowe wymagania i odpowiada warunkom zawartym w dyrektywie 1999/5/EC
Portuguese	[AudioCodes Ltd] declara que este [Mediant 1000 Digital Series] está conforme com os requisitos essenciais e outras disposições da Directiva 1999/5/CE.
Slovak	[AudioCodes Ltd] týmto vyhlasuje, že [Mediant 1000 Digital Series] spĺňa základné požiadavky a všetky príslušné ustanovenia Smernice 1999/5/ES.
Slovene	Šiuo [AudioCodes Ltd] deklaruoja, kad šis [Mediant 1000 Digital Series] atitinka esminius reikalavimus ir kitas 1999/5/EB Direktyvos nuostatas.
Spanish	Por medio de la presente [AudioCodes Ltd] declara que el [Mediant 1000 Digital Series] cumple con los requisitos esenciales y cualesquiera otras disposiciones aplicables o exigibles de la Directiva 1999/5/CE
Swedish	Härmed intygar [AudioCodes Ltd] att denna [Mediant 1000 Digital Series] står I överensstämmelse med de väsentliga egenskapskrav och övriga relevanta bestämmelser som framgår av direktiv 1999/5/EG.

Safety Notices

1. Installation and service of this gateway must only be performed by authorized, qualified service personnel.
2. The equipment must be connected by service personnel to a socket-outlet with a protective earthing connection.
3. This equipment should be installed in restricted access locations with maximum allowed temperature 40°C (104°F)
4. To avoid risk of fire use 26 AWG or higher wiring to connect the telecom ports.

Mediant 1000 iPMX Notice

When an iPMX (Pentium) module is used, to comply with Class B limits, a shielded cable should be used for the Ethernet port of the iPMX.

ACTA Customer information

1. This equipment, the VoIP Gateway, models Mediant 1000 complies with Part 68 of the FCC Rules and the requirements adopted by the ACTA. On the bottom of the unit of this equipment is a label that contains among other information, a product identifier in the format US:AC1ISNANM1KDIG. If requested, this number must be provided to the telephone company.

2. This equipment is designed to be connected to the telephone network using an RJ-48C connector, which is Part 68 compliant. The service order code (SOC) is 6.0F and the Facility interface codes (FIC) are: 04DU9.1SN, 04DU9.1KN, 04DU9.BN, 04DU9.DN.

3. Should the product causes harm to the telephone network, the telephone company will notify you in advance that temporary discontinuance of service may be required. If advance notice is not practical, you will be notified as soon as possible. Also, you will be advised of your right to file a compliant with the FCC if it is necessary.

4. The telephone company may make changes in its facilities, equipment, operations or procedures that could affect the operation of the equipment. If this happens, the telephone company will provide advance notice in order for you to make necessary modifications to maintain uninterrupted service.

5. If trouble is experienced with this equipment, for repair or warranty information please contact AudioCodes Inc., 2099 Gateway Place, Suite 500, San Jose, CA, 95110, phone number 1-408-441-1175. If the equipment is causing harm to the telephone network, the telephone company may request to disconnect the equipment until the problem is resolved.

6. Installation is described in Product installation manual. Connection to Telephone Company-provided coin service is prohibited. Connection to party lines service is subject to State tariffs.

Network Information and Intent of Use

The products are for access to ISDN at 2048 kb/s and for access to G.703 Leased lines at 2048 kb/s.

Network Compatibility

The products support the Telecom networks in EU that comply with TBR4 and TBR13.

Telecommunication Safety

The safety status of each port is declared and detailed in the table below:

Ports	Safety Status
E1 or T1	TNV-1
Ethernet (100 Base-TX)	SELV

TNV-1: Telecommunication network voltage circuits whose normal operating voltages do not exceed the limits for SELV

SELV: Safety extra low voltage circuit.

FCC Statement

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

Reader's Notes

AudioCodes™ Mediant™ Media Gateways

SIP

Mediant 1000

User's Manual Version 5.0

CERTIFIED QMS
SI
ISO 9001:
2000
THE STANDARDS INSTITUTION OF ISRAEL

AudioCodes

http://www.audiocodes.com/